NR INDEKSU 353965
PL ISSN 0860-1674

3

Z MIKROKOMPUTEREM NA TY

Bajtek

MAGAZYN KOMPUTEROWY

NR 3 (61) '91 CENA 7600

Jak złożyć (tani) komputer?



TELEMONITOR TMP 205

DRUKARKA D-100 MPC

program KAT 90

KOSTKA CIĘŻARKA

KOMPUTERY W LO

TIMEX I AUTOFIRE

Kiedy będzie Wielkanoc?

1581 KŁOPOTÓW

DRUKARKA STAR LC-20

— to nowa, szybsza LC-10



- Prędkość druku: 180 zn./sek.
- Jakość druku: standard oraz NLQ
- Traktor pchający
- „Parkowanie" papieru
- Automatyka oddzierania papieru
- Interfejs Centronics

Cena 2.500.000 (orientacyjna cena detaliczna)

MULTI-FONT A B C D

PAPER PARK FUNCTION P

SPEED 180 cps 12 cpi

DRUKARKA LC-200

— Star znów ustanawia nowy standard!

- Max. prędkość druku: 225 zn./sek.
- Druk kolorowy
- Możliwość podawania papieru od dołu
- Traktor pchający i ciągnący
- „Parkowanie" papieru
- Automatyka oddzierania papieru
- Interfejs Centronics

Cena 3.900.000
(orientacyjna cena detaliczna)

PAPER PARK FUNCTION P



star Twoja drukarka

MULTI-FONT A B C D

HIGH SPEED FEATURE 225 cps 12 cpi

BOTTOM FEED

COLOUR PRINTING

PUSH AND PULL TRACTOR

ABC DATA WARSZAWA

Przedstawicielstwo w Polsce
ABC Data Warszawa
ul. Waliców 13

tel. 24-11-43
24-78-35
tx. 816-423

Jadłospis

KOMPUTER '91

Wydarzeniem stycznia dla osób zainteresowanych sprzętem i oprogramowaniem była coroczna, warszawska wystawa Komputer'91, organizowana przez firmę Agpol. Rozmiarami i wystrojem dominowały stoiska dużych i znanych firm zachodnich.

Byli też starzy znajomi. Włoska firma Neolt, specjalizująca się w sprzęcie biurowym, oferowała, jak co roku rewelacyjny miniploter dla kreślarza. W jakimś dobrze zakamuflowanym stoisku można było odnaleźć miniaturowe komputery CARRY-I firmy Flytech — niestety tylko modele serii 8000. Te nowsze, wyposażone w procesor 386, jeszcze do nas nie dotarły. Ciekawostką techniczną był 3.5"-calowy napęd firmy TEAC o pojemności 2.8 MB, stosowany m.in. w najnowszych komputerach NeXT.

W dziedzinie sprzętu nie było w zasadzie większych rewelacji, natomiast, jeśli chodzi o oprogramowanie, to sytuacja w stosunku do lat ubiegłych uległa poprawie. Polskie firmy software'owe opanowały proces instalacji naszego alfabetu narodowego w komputerach i drukarkach, a prezentowane oprogramowanie nie straszyło napisami w rodzaju „zadanie kata na lace".

Również zachodni producenci zjawili się w gronie liczniejszym niż dotychczas. Posiadanie legalnego oprogramowania — powoli i z pewnymi oporami — wchodzi jednak u nas w modę. Z tego powodu znacznie łatwiej można nabyć licencjonowane programy, ale niestety w cenie często dwukrotnie większej niż na Zachodzie.

Na wystawie sporo firm miało w swojej ofercie dużą liczbę coraz popularniejszych komputerów typu laptop. Ceny zaczynały się już od 1300 USD za XT z dyskiem 20MB. Sądzę, że przy odpowiedniej promocji sprzęt tej klasy znalazłby duże uznanie w oczach wielu potencjalnych nabywców. Sam od dłuższego czasu posługuję się historycznym już modelem BW8, oferowanym parę lat temu przez CSH. Ponieważ jest on wyjątkowo powolny i ciężki, coraz częściej myślę o jakimś lżejszym sprzęcie typu notebook, wyposażonym w procesor i386. W niedalekiej przyszłości, dzięki nowej, rewelacyjnej kości AM 386 DXL produkowanej przez amerykańską firmę Advanced Micro Devices, będzie ich znacznie więcej. Nowy układ, w stosunku do produktu firmy Intel, wyróżnia się trzykrotnie mniejszym zużyciem mocy i większą szybkością, przy stu procentowej zgodności z pierwowzorem. Komputer wyposażony w procesor AM 386 DXL pracujący z zegarem 40 MHz jest szybszy od sprzętu opartego na 33-megahercowym i486.

Nasze komputery — nawet te opisywane w klanie IBM, gdzie znajdą Państwo odpowiedź, jak złożyć taniego „peceta" — są znacznie wolniejsze. Nie stanowi to specjalnego problemu przy grach — patrz program Kostka Ciężarka.

Oprócz Klanu IBM i innych prezentujemy w aktualnym numerze kilka testów sprzętu i oprogramowania rodzimej produkcji. A w ogóle namawiamy do czytania całego Bajtka od pierwszej do ostatniej strony. Aby ułatwić Państwu to zadanie, wprowadzamy od kwietnia comiesięczny konkurs, który będzie stanowił sprawdzian znajomości materiału zawartego w całym numerze. Lista nagród do wylosowania zostanie podana w Bajtku 4/91.

Na zakończenie przeprosiny wobec Czytelników, którzy długo czekali na numer styczniowy. Ze względu na zmiany w objętości i problemy techniczne drukarni w Ciechanowie, dotarł on do kiosków dopiero w połowie lutego, wyprzedzając numer 2/91 tylko o dwa tygodnie. Prosimy o wyrozumiałość — od numeru marcowego będziemy nadganiać powstałe opóźnienie.

Jarosław Młodzki

PRENUMERATA

Przyjmujemy wpłaty na prenumeratę wszystkich wydawanych przez nas pism. Prenumerata rozsyłana jest pocztą. Wpłaty prosimy dokonywać na nasze konto:

Spółdzielnia „Bajtek"
Bank „Agrobank S.A."
479994-1834-131
ul. Grochowska 262
04-398 Warszawa

Po dokonaniu wpłaty należy wysłać na adres redakcji kopię przekazu oraz dokładne dane prenumeratora: pełny adres, informację o tytułach na które została dokonana wpłata (oraz ewentualnie numer prenumeraty).

Pierwsze możliwe numery czasopism wyślemy w ciągu 6—8 tygodni od otrzymania informacji. Oferta jest ważna do 30.IV.91.

tytuł \ ilość zeszytów	3	6	12
Bajtek	X	45600	91200
Moje Atari	18000	36000	X
Top Secret	21600	43200	X

Bajtek
MAGAZYN KOMPUTEROWY

Redakcja:
ul. Wspólna 61,
00-687 Warszawa,
tel. 21-12-05

Redagują:
Redaktor naczelny — Jarosław Młodzki
Sekretarz redakcji — Marcin Przasnyski
Opracowanie graficzne — Wanda Roszkowska
Zdjęcia — Leopold Dzikowski
Szefowie klanów:
Amstrad — Jonasz Mayer
Atari — Wojciech Zientara
Commodore — Klaudiusz Dybowski
IBM — Marcin Borkowski
Spectrum — Maciej Pietraś
Co jest grane — Łukasz Czekajewski
Edukacja — Tadeusz B. Mańk
Micro Magazyn — Janusz Jarmoch
Stali współpracownicy:
Grzegorz Bujanowski
Jarosław Burczyński
Marek Czarkowski
Piotr Kos
Robert Magdziak
Waldemar Nowak
Andrzej Pilaszek
Mieczysław Płacheta
Maria Radzimińska
Marek Sawicki
Piotr Sumara
Michał Szokoło
Stanisław Winiecki

Wydawca:
Spółdzielnia „Bajtek"
ul. Wspólna 61
00-687 Warszawa
Skład i druk:
Prasowe Zakłady Graficzne w Ciechanowie
Fotoskład: Grażyna Kurzątkowska
Montaż: Grażyna Ostaszewska
Korekta: Maria Krajewska
Teresa Rutkowska
Nakład: 107 tys. egz. Zam. 84260.

Celem ułatwienia zainteresowanym kontaktów z załogami poszczególnych klanów, stworzyliśmy system dyżurów szefów klanów. Najskuteczniejsze będzie dzwonienie w podanych godzinach w następujące dni:
Amiga — piątek 15.00 — 17.00
Amstrad — piątek 16.00 — 18.00 tel. 284594
Atari (Moje Atari) — wtorek, środa, czwartek 10.00 — 14.00
Co jest grane (Top Secret) — środa 12.00 — 16.00
Commodore — środa, 15.00 — 18.00
IBM — czwartek 15.00 — 18.00
Po dzwonku — wtorek 12.00 — 16.00
Spectrum — wtorek 14.30 — 16.00



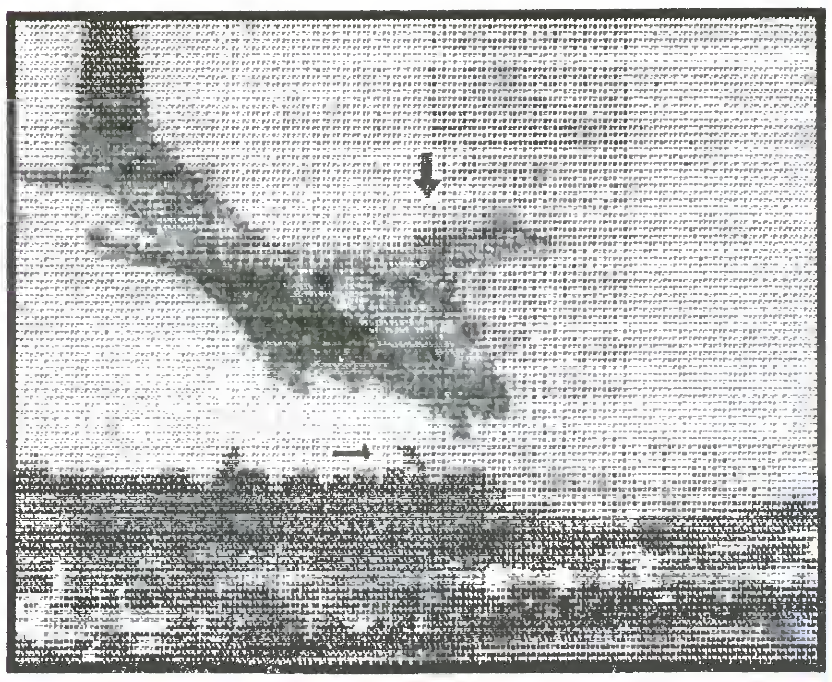
MANIPULACJA...

Taki „wstępniak" znalazłby się prawdopodobnie w jakiejś gazecie codziennej przy tym zdjęciu.
W rzeczywistości tragedia wydarzyła się w górnych obszarach pamięci RAM komputera typu XT, do którego podłączyliśmy nasz prosty redakcyjny scanner. „Retuszu" dokonaliśmy usuwając po prostu pewne fragmenty zdjęcia oryginalnego, w celu wzmożenia akcentu dramatycznego. Dla porównania zamieszczamy również zdjęcie oryginalne.

Osobiście możliwości takie trochę mnie niepokoją. Komputer przestaje już powoli być maszyną tak zupełnie niewinną, jakby się to mogło wydawać. Większość takich komputerowych fałszerstw popełnianych jest przede wszystkim w prospektach reklamowych, składanych i wydawanych za pomocą komputera. Bez względu na to, czy jest to oszustwo małe czy duże, to jednak jest to fałszerstwo. Komputer nie ma moralności i trudno go za to winić; pytanie tylko, czy taka maszyna w rękach człowieka pozbawionego jakichkolwiek norm nie przedzierzgnie się automatycznie z niewiniątka w morderczy wynalazek.

(KD)

„Tragedia ta wydarzyła się podczas pokazów lotniczych. Podchodzący do lądowania samolot typu „Buffalo" nagle zboczył z kursu, pochylił ostro dziób do dołu i rozbił się na przylegającym do lotniska parkingu, pełnym w tym czasie ludzi i pojazdów. Przyczyną katastrofy było zmęczenie materiału klap, których nienormalną pozycję zaznaczono strzałką pionową; druga strzałka wskazuje na dezintegrację fragmentów klapy tuż przed wykonaniem zdjęcia. Dokument ten został wykonany przez przypadkowego fotografa — amatora".





Wszyscy żeglarze z utęsknieniem czekają na rozpoczęcie kolejnego sezonu. Ale pożeglować można także zimą, i to nie ruszając się z domu. Program PC-Navigator 2 pozwala jeszcze raz przeżyć wakacyjne przygody i przy okazji poszerzyć wiedzę z dziedziny nawigacji. Do jego uruchomienia wystarczy najprostszy komputer PC ze stacją dysków 5.25" lub 3.5", 512 KB pamięci RAM i kolorowym monitorem.

Żegluga odbywa się na mapie przeniesionej z prawdziwych nawigacyjnych map brytyjskiej admiralicji. Obok fragmentu akwenu widzimy na ekranie sylwetkę jachtu oraz przyrządy nawigacyjne. Odczytujemy z nich kurs kompasowy, stan logu, głębokość, kierunek i prędkość wiatru, ciśnienie atmosferyczne, stan silnika.

PC-Nawigator może zasymulować szereg czynności, które wykonuje się

AccuCard — miniaturowy UPS



Przydatność UPS-ów, jako dodatkowego zabezpieczenia szczególnie cennych danych nie jest przez nikogo kwestionowana. Niestety wysoka cena i spore rozmiary tych urządzeń odstraszają potencjalnych nabywców.

W tym kontekście bardzo ciekawą ofertę stanowi produkt firmy Emerson Electric o nazwie AccuCard. Jest to miniaturowy UPS dostarczany w formie dodatkowej karty do komputera IBM PC. Urządzenie to zawiera między innymi baterię akumulatorów pozwalającą na kilkuminutowe działanie komputera po wyłączeniu zasilania sieciowego. Specjalne oprogramowanie „zaszyte" w karcie pozwala na dokładne zapamiętanie stanu procesora i pamięci w przypadku awarii napięcia i odtworzenie tego stanu przy ponownym włączeniu komputera.

Z punktu widzenia użytkownika działanie tej karty jest zupełnie przezroczyste. Wyłączając nawet przypadkowo komputer w trakcie pisania tekstu, przy ponownym włączeniu znajdziemy się w tym samym miejscu, gdzie ostatnio skończyliśmy pisać.

Wspomniane urządzenie nie zajmuje dodatkowego miejsca na biurku, kosztuje 179 funtów i dzięki wbudowanemu oprogramowaniu działa automatycznie.

(JM)

DYSKIETKI 20 MB

Typowe dyskietki elastyczne mają pojemność od 170 KB do 1,44 MB. Dla wielu użytkowników to zdecydowanie za mało. Nowoczesne pakiety oprogramowania zajmują niejednokrotnie po kilka megabajtów. W tej sytuacji najlepiej zaopatrzyć się w twardy dysk, ale i on zapełni się po jakimś czasie. Wymieniony nośnik informacji jest niezbędny także ze względu na konieczność robienia kopii zapasowych najcenniejszych zbiorów.

Kłopoty związane z przechowywaniem danych rozwiązują oczywiście dyski wymienne o dużej pojemności, streamery kopiujące zawartość dysku na taśmę, lub dyski optyczne. Są to niestety dosyć drogie urządzenia, na które nie każdy może sobie pozwolić. Nie należy więc dziwić się, że wiele firm pracuje nad zagadnieniem zwiększenia pojemności zwykłych napędów dysków elastycznych.

Jedno z rozwiązań opracowała firma Verbatim. Nośnik przypomina zwykłe dyskietki 3,5", lecz ma wielkość 5,25". Plastikowa obudowa i metalowa, odsuwana blaszka zapobiegają przedostaniu się szkodliwych zanieczyszczeń do wnętrza. Producent oferuje dwie wersje napędu. Stacja dysków Verbatim 12 pozwala uzyskać po sformatowaniu dyskietki 9,7 MB, a Verbatim 24 — 20 MB.

(J)

Na podst. St Computer

PC NAVIGATOR



w prawdziwej żegludze. Najistotniejszą z nich jest określenie pozycji. Służy do tego między innymi Decca, system nawigacji radiowej. Urządzenia tego systemu są dość drogie i do dziś jeszcze nie wyposażono w nie wszystkich jachtów. Elektroniczna symulacja daje więc wielu żeglarzom okazję podniesienia kwalifikacji.

Pozycję można również określać w tradycyjny sposób, z pomiarów położenia obiektów o znanych współrzędnych geograficznych. Na ekranie widoczny jest wtedy trójwymiarowy obraz. Przy pomocy klawiszy kursora możemy odszukać latarnie, nabieżniki, boje, a następnie wykreślić pozycję na mapie.

Program uwzględnia także symulację warunków atmosferycznych i pływów. Możliwy jest także trening odbioru prognozy pogody i wyświetlenie mapy synoptycznej. Przy pogarszających się warunkach atmosferycznych żeglarze muszą odpowiednio zmieniać żagle.

Producent przewiduje użytkowanie programu PC-Navigator w trzech sytuacjach: na kursach nawigacji, jako narzędzie pomagające w samodzielnym podnoszeniu umiejętności żeglarskich i ewentualnie jako gra symulacyjna. Programu nie można używać w rzeczywistej żegludze, choć pewne jej elementy oddane są bardzo wiernie.

Na podst. Personal Computer World

1. Mapa pogody w programie PC-Navigator.

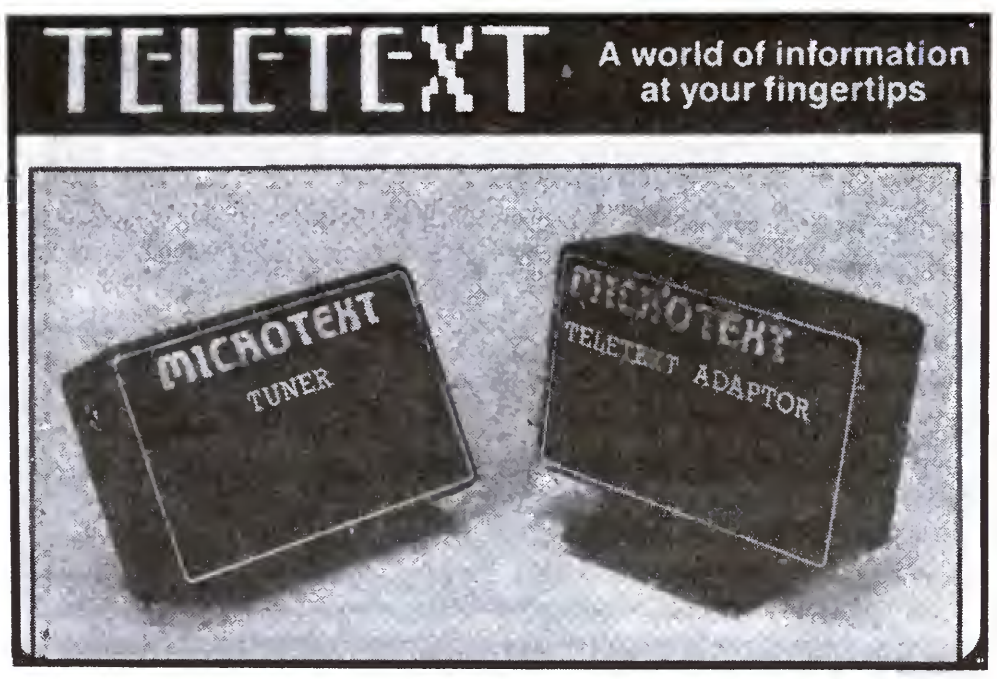
2. Symulacja odbiornika w systemie nawigacji radiowej Decca.

TELETEXT NA COMMODORE

Jeżeli chcemy poznać najnowsze wiadomości, wyniki niedawnego meczu lub przeczytać reklamę o rewelacyjnym środku na porost pierza u trzytygodniowych brojlerów, a w pobliskim kiosku zabrakło gazet, natomiast w telewizji z jakiegoś powodu od godziny pokazywana jest plansza *Spokojnie!To tylko awaria*, jedynym wyjściem jest telegazeta. Niestety ci, którzy nie posiadają telewizora przystosowanego do odbioru teletextu lub chociażby dekodera pozwalającego na odbiór telegazety pozostają jedynie ze świadomością, że coś takiego istnieje. Wiele do życzenia pozostawia także cena samego dekodera na zapłacenie której nie każdy może sobie pozwolić.

Przynajmniej częściowym rozwiązaniem (niestety tylko dla posiadaczy Commodore 64/128) może być interfejs firmy MICROTEXT zmieniający poczciwy komputer w odbiornik teletextu. W komplecie znajduje się moduł oraz tuner włączony pomiędzy komputer a antenę telewizyjną. Zaletą tego interfejsu jest prostota obsługi oraz niewysoka cena. Na razie dostępny jest na zachodnim rynku, ale wkrótce mamy szansę zobaczyć go w Polsce.

(TNL)



KOMPUTERY W ZSRR

„Pieriestrojka" mimo kłopotów (brak funduszy) i ograniczeń (COCOM) dotknęła także radziecki rynek sprzętu komputerowego. Już to „Polskije pany", już to „Kitajcy" i przedstawiciele innych zaprzyjaźnionych nacji ruszyli sprzedawać narodom ZSRR wszystko, co jest lepsze od IBM PC AT, z dyskiem twardym 40 MB i kartą graficzną EGA.

Rozmiary zjawiska stały się na tyle znaczne, że największy magazyn komputerowy świata „Byte" wyprawił w moskiewską stronę swego brytyjskiego współpracownika Martina Banksa, aby na miejscu sprawdził, czym się w tym sezonie handluje w „ojczyźnie światowego bolszewizmu".

Shirkant H. Joshi z firmy Tandon zapewniał, że robi z Rosją dobre interesy, korzystając ze strefy wolnocłowej w Bombaju. Komputery klasy IBM PC idą jak ciepłe bułeczki, ograniczeniami zaś nikt się nie przejmuje.

Jeszcze bardziej pomysłowy okazał się dr Philip Hwang, prezes firmy TeleVideo Systems z San Jose w Kalifornii, który wszedł w kilka spółek „dżonwencer" z tubylcami, między innymi z Ministerstwem Rolnictwa.

Z informacji Banksa wynika, że i wielcy nie zasypiają gruszek w popiele. Przedstawicielstwo IBM otwarto w Moskwie już w roku 1974. Siemens w tym roku podpisał kontrakt na dostawę do ZSRR 200 tysięcy komputerów klasy IBM PC. Skandynawski koncern Nokia penetruje od dawna rynek radziecki, podobnie jak włoska firma Olivetti i Amstrad, który ostatnio dostarczył Borysowi Jelcynowi jeden z modeli PCW, aby ten „Robin Hood Pierestrojki" miał na czym spisać swoje memuary.

Microsoft w porozumieniu z miejscowymi programistami wprowadził na rynek radziecką wersję systemu operacyjnego MS-DOS. Apple zaś proponuje cyrylicę dla komputerów Macintosh. Wkrótce też w ZSRR rozpocznie się produkcja komputerów klasy IBM PC.

Hasło „Go east young man" znane jest przedstawicielom zachodniego biznesu od dawna. Mało osób wie, że już w 1927 roku stary Henry Ford sprzedał Krajowi Rad fabrykę ciężarówek o zdolności produkcyjnej 100 tysięcy pojazdów rocznie. Na początku lat siedemdziesiątych jego wnuk — także Henry — powtórzył kontrakt dziadka. Z komputerami jest podobnie.

Bream



JAK TO ROBIĄ INNI?

rozmowa z dr. Tomem deFranco z Uniwersytetu Hartford, USA

— Czy mógłby Pan powiedzieć, jak w Stanach Zjednoczonych wykorzystywane są komputery w szkolnictwie?

— Kiedyś komputery w szkołach były wykorzystywane tylko do nauki programowania. Nauczano głównie BASIC'a. Dziś wprawdzie także uczymy programowania, ale w większości jest to LOGO lub LOGOWRITER — wersja LOGO z możliwością edycji tekstów, dźwięku, sterowania urządzeniami zewnętrznymi itp. Języki te wybraliśmy ze względu na ich walory dydaktyczne. Następnie komputer został oddany w ręce ucznia, któremu umożliwiono bezpośrednią styczność ze sprzętem. Komputer zadawał pytania, uczeń wprowadzał odpowiedź, program mówił „dobrze" lub „źle" i to było wszystko. Ale że ta droga nie jest dobra zauważono już **25 lat temu** (podkr. TBM). Dziś uznajemy, że dobrą drogą jest idea nauczania obsługi programów. Powstało wiele programów użytkowych i nauka ich zastosowania jest naszym podstawowym celem. LOGO uczymy tylko ze względu na jego przydatność w matematyce. Stosujemy również często system LEGO-LOGO. Jest to połączenie klocków LEGO z zespołem silniczków sterowanych przez komputer. Dzięki tej pomocy nauczać można nie tylko robotyki, ale także nauk przyrodniczych, informatyki, a nawet matematyki.

Niezmiernie ważne jest nauczenie posługiwania się **Edytorami Tekstów**, **Bazami Danych** czy **Arkuszami Elektronicznymi**. Te programy wprowadzamy w czwartej-piątej klasie. W naukach przyrodniczych wykorzystujemy również komputer w laboratorium szkolnym. Połączony z odpowiednimi czujnikami mierzy ruch, światło, temperaturę, a wyniki doświadczeń uczniowie muszą opracować sami, wykorzystując znane sobie programy.

Zawsze staramy się wykorzystać komputer jako **narzędzie**, nie jako wartość samą w sobie. Programy nauczania są dostosowane do możliwości wykorzystania komputerów, a nauczyciele mają dużo swobody.

— Czy macie również programy stworzone specjalnie dla szkół?

— Tak. Mamy wiele krótkich programów napisanych dla rozwiązywania konkretnych problemów na lekcjach różnych przedmiotów. Programy te są dostępne dla nauczyciela, który sam decyduje, czy i kiedy je wykorzystać.

— Kto tworzy te programy? Kto za nie płaci?

— Są one tworzone na uniwersytetach przez profesjonalistów — dydaktyków i informatyków. Ale także wiele z nich zostaje napisanych przez studentów czy nauczycieli — uczestników kursów. Powstają także programy tworzone przez firmy software'owe na zamówienia uniwersytetu czy rządu. Taki program raz kupiony jest później rozpowszechniany w szkołach bezpłatnie, jako „public domain".

— Jak uczniowie przyjmują komputery?

— Są zachwyceni.

— Wszyscy?

— Raczej tak. Zdecydowana większość lubi takie lekcje, są chętni do nauki i bawią się doskonale. Początkowo nauczyciele uczą ogólnej kultury informatycznej, posługiwania się sprzętem, ale szybko się okazuje, że mogą się od uczniów sami niejednego nauczyć.

— W jakie komputery wyposażone są szkoły amerykańskie?

— Głównie w Apple II. Ten typ przeważa, choć zdarzają się i inne. Jeśli chodzi o ilość sprzętu, to rozpiętość jest ogromna. Obok bogatych szkół wyposażonych lepiej niż niejeden uniwersytet mamy i takie, w których jest zaledwie kilka komputerów.

— Czy w USA szkoły połączone są ze sobą siecią telekomunikacyjną?

— Takie próby istnieją. Siecią przeznaczoną specjalnie dla szkół jest np. sieć National Geographic, w której uczniowie zbierają dane dotyczące pogody, skażenia środowiska, dane socjologicznie. Prócz tego istnieje wiele sieci o zasięgu lokalnym łączących szkoły z uniwersytetem stanowym. Tą drogą wymieniane są informacje i nowinki, a także przesyłane są programy edukacyjne, o których mówiliśmy wcześniej.

— Czy macie specjalne programy kształcenia nauczycieli do pracy z komputerami?

— Raczej nie. Mój uniwersytet prowadzi dla nauczycieli szkół podstawowych dodatkowy kurs wykorzystania komputerów, ale robimy to na własną rękę. Nie ma ogólnokrajowego programu, choć oczywiście w trakcie zwykłego szkolenia prezentowane są możliwości wykorzystania sprzętu i oprogramowania w szkole.

— Jak, Pańskim zdaniem, powinna rozwijać się komputeryzacja szkolnictwa w Polsce?

— To bardzo trudne pytanie. Odpowiedzi musicie sobie sami udzielić, ale wydaje mi się, że trzeba najpierw rozwiązać trzy problemy. Po pierwsze, musicie mieć sprzęt (komputery i oprogramowanie). Po drugie, musicie mieć wykształconą kadrę — ludzi, którzy wiedzą, jak wykorzystać komputer w szkole, by przynosiło to jak największe korzyści. Po trzecie wreszcie, musicie odpowiedzieć sobie na pytanie, jak w Polsce będą wykorzystywane komputery poza szkołą. Powinniście przygotować uczniów do przyszłej pracy zawodowej. Musicie mieć także nauczycieli, którzy wiedzą, jak wykorzystać komputery i czego uczyć, by w pełni wykorzystać ich możliwości. Krótko mówiąc — potrzebujecie ludzi, sprzętu i pieniędzy.

— Dziękuję za rozmowę.

T.B. Mańk



KOMPUTERY w LO

W latach 1989—1990 prowadzone były[1)] badania nad stanem wykorzystania komputerów w szkolnictwie średnim w Polsce.

Badaniom poddane było wyposażenie szkół w sprzęt komputerowy i oprogramowanie dydaktyczne, sposoby wykorzystania komputerów w szkole, trudności napotykane przy stosowaniu komputerów, postawy nauczycieli i in.

W artykule tym postaramy się przedstawić najważniejsze, naszym zdaniem, wyniki tych badań.

1. Typy komputerów

Wśród typów komputerów używanych w szkolnictwie zdecydowanie króluje ZX-Spectrum. Ten typ posiada 89% szkół wykorzystujących komputery, co razem ze zgodnym z nim ELWRO 800 — Junior (46%) daje mu dużą przewagę. Niespodzianką jest silna pozycja SVI (13%) i IBM (12%). Inne komputery (Amstrad, Commodore, Atari, Meritum i inne) posiada 36% szkół.

Gdy na te same dane spojrzymy z innego punktu widzenia i policzymy, ile jakiego sprzętu znajdowało się w szkołach, to okaże się, że wprawdzie Spectrum (37% komputerów w szkołach) i Junior (42%) nadal zajmują pierwsze miejsce, ale SVI (8%) jest prawie tyle samo co innych (10%), a prawie trzy razy więcej niż IBM (3%).

2. Urządzenia peryferyjne

O jakości sprzętu komputerowego w znacznej mierze decydują urządzenia peryferyjne: monitory, drukarki, pamięci zewnętrzne i inne. Jeśli chodzi o pamięci masowe, to sytuacja jest wręcz dramatyczna. **Ponad połowa komputerów wykorzystuje jako pamięć zewnętrzną taśmy magnetofonowe.**

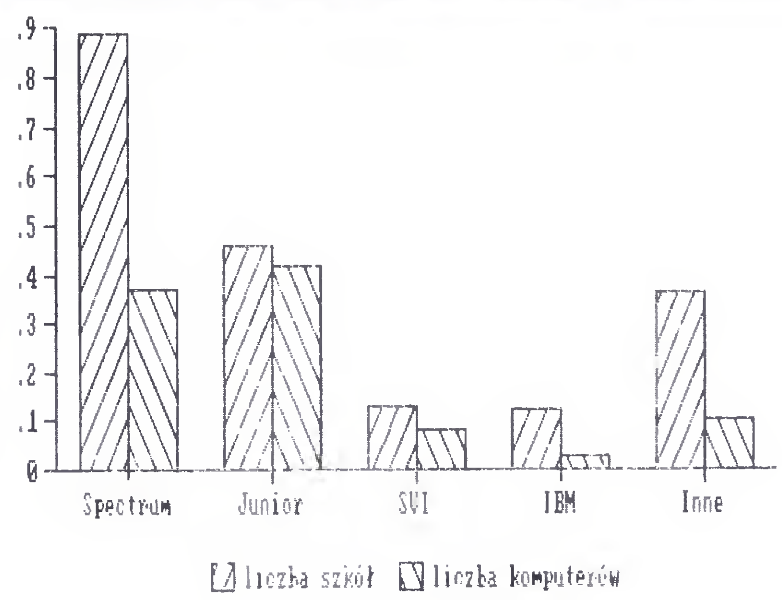
Drukarkami dysponowało 79% szkół wykorzystujących komputery, a kolorowym monitorem lub telewizorem tylko 38%.

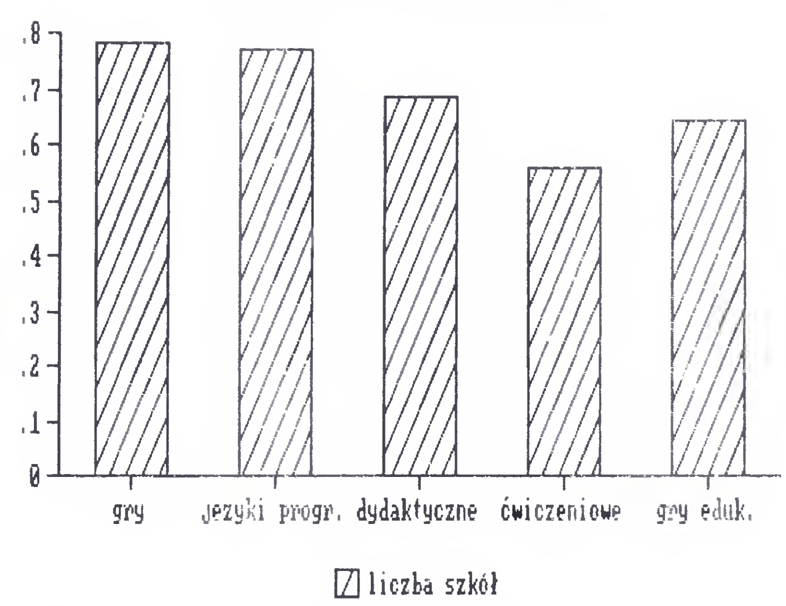
3. Lokalizacja sprzętu w szkole

W szkole komputery znajdują się głównie w pracowni przeznaczonej do nauczania podstaw informatyki (84,7% szkół). Tylko 22,4% szkół posiada na stałe komputery w innych pracowniach, z czego 12% pełni zastępczo rolę pracowni informatycznej. Zaledwie w 7% szkół komputery są umiejscowione w więcej niż dwóch pracowniach.

4. Oprogramowanie

O rodzaju nabywanego oprogramowania najczęściej decydowali nauczyciele koordynujący wykorzystanie komputerów (49%). Wpływ na ilość i oprogramowanie miały również: kuratorium (33,3%), pojedynczy nauczyciel (32,5%) i dyrektor (25,9%). W efekcie prawie połowa szkół nie posiada edytorów tekstu (zresztą bez drukarek są one i tak niewiele przydatne), więcej niż połowa nie dysponuje żadną bazą danych, a programy kalkulacyjne posiada mniej niż 20% szkół. Na czele listy najczęściej posiadanych programów znajdują się... gry rozrywkowe! (78,2%) i języki programowania (77,3%). Na szczęście w wielu szkołach można znaleźć programy nauczające (69%), ćwiczeniowe (56%), gry dydaktyczne (64,2%), choć w tym ostatnim przypadku nie wiadomo, ile z nich jest naprawdę dydaktycznych.





Najwięcej jest programów pomocnych w nauczaniu podstaw informatyki (dostępne w 82% szkół). Matematyk może wspomóc lekcję komputerem w 74% szkół, a programy do nauczania przedmiotów przyrodniczych (fizyka, chemia, biologia i geografia łącznie) posiada 65% szkół. Najmniej programów ma do dyspozycji nauczyciel przedmiotów społecznych.

VMATGRAF-



KOMPUTEROWA TABLICA

Komputery w szkole można stosować na wiele sposobów. Animacja, symulacje czy modelowanie — to metody nauczania dostępne tylko przy jego użyciu. Ale komputer jest narzędziem na tyle uniwersalnym, że z powodzeniem może zastąpić inne używane w szkole pomoce dydaktyczne, również tak niezastąpione jak tablica.

W nauczaniu matematyki często pojawiają się wykresy. Wykres funkcji bardziej pomoże uczniowi zrozumieć jej charakter, niż długi i skomplikowany dowód. Ale nie każdy nauczyciel matematyki jest obdarzony talentem artystycznym. Wykresy są więc często schematyczne i zniekształcone.

Autorzy programu **VMATGRAF** starali się pomóc nauczycielom matematyki w ich zmaganiach z kredą i tablicą. Dzięki niemu nauczyciel zostaje odciążony od konieczności rysowania wykresów, żmudnego i nieprecyzyjnego odczytywania danych oraz wykonywania obliczeń rachunkowych.

Główne zastosowania programu to możliwość rysowania dowolnych funkcji, ciągów oraz krzywych opisanych równaniami parametrycznymi. Wykresy te można później badać dzięki możliwości odczytania położenia punktów, rysowania brzegów zbiorów zawierających punkty, których współrzędne należą do dowolnych otoczeń, rysowania ilorazów różnicowych, rysowania siecznych itp.

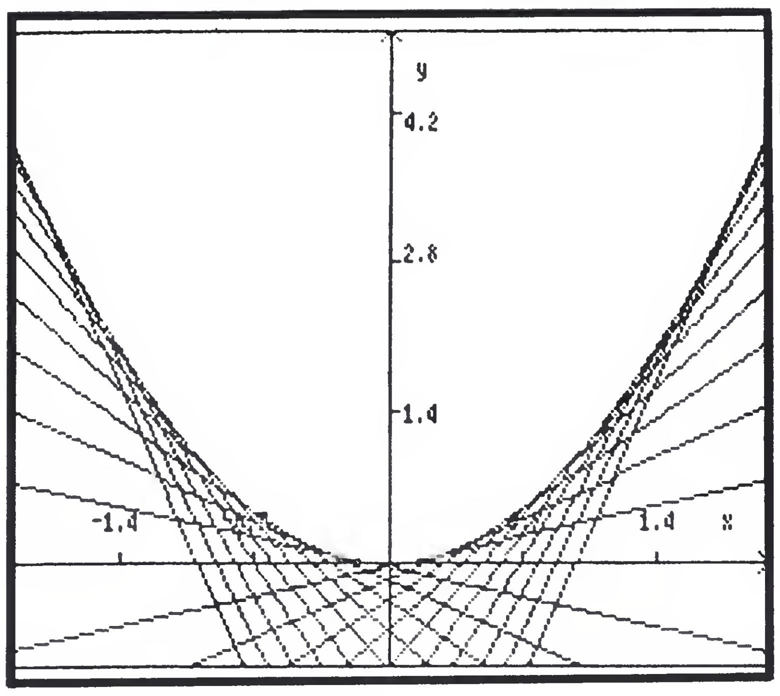
VMATGRAF nie jest programem nauczającym. Jest to program pomocniczy i jako taki spełnia swoje zadanie bardzo dobrze. Daje nauczycielowi nie tylko możliwości przedstawienia przebiegu zmienności funkcji, ale także możliwość prezentacji ciągów liczbowych, ich zbieżności i granicy, przekształceń i przesunięć symetrycznych wykresów i sporo innych.

Przedstawione obok rysunki pozwalają wyrobić sobie własne zdanie o możliwościach programu. Czy, jak i kiedy go stosować, każdy nauczyciel musi sam wiedzieć.

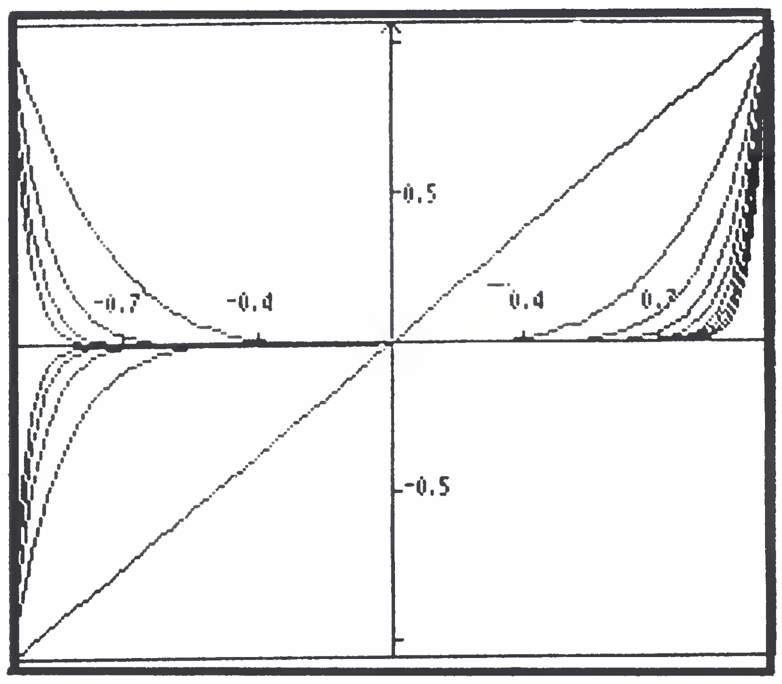
Program, jak na warunki polskie, jest napisany z dużą dbałością o wygodę użytkownika. Obsługa odbywa się głównie przy użyciu kursorów (poza wprowadzaniem danych oczywiście), a we wszystkich opcjach znajdują się pożyteczne dodatki ułatwiające pracę (np. zmiana wybranego parametru kursorem, rysowanie rodziny krzywych i inne). Warto podkreślić istnienie dość obszernej i dokładnej instrukcji obsługi, co w warunkach polskich jest raczej rzadkością.

Dziennikarz opisujący efekty czyjejś pracy jest zawsze w kłopotliwej sytuacji. Gdy pisze dobrze, może być posądzony o kryptoreklamę, gdy źle — o zawiść. Opisany program podobał nam się i nie możemy tego nie napisać. Zgodnie z naszą najlepszą wiedzą, jest on wart polecenia.

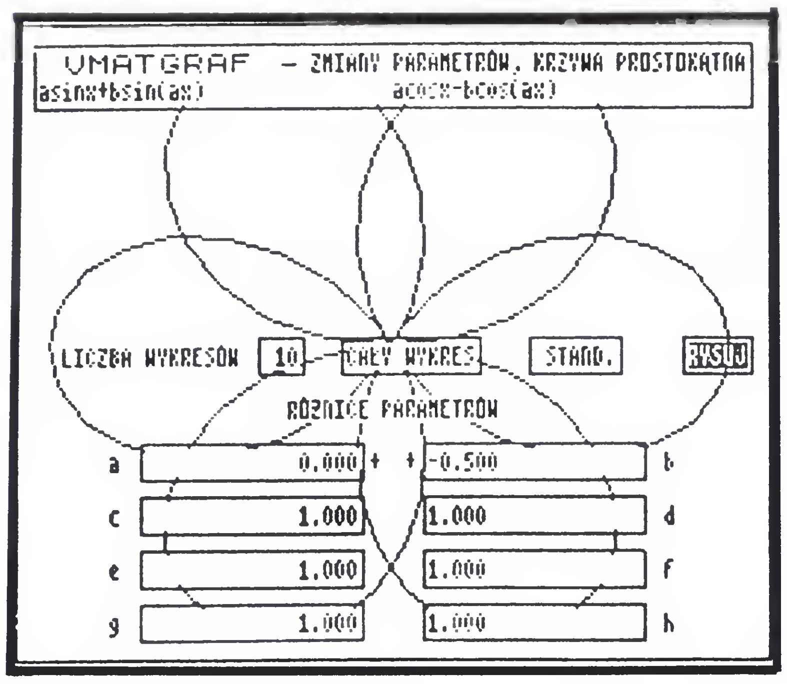
T.B. Mańk



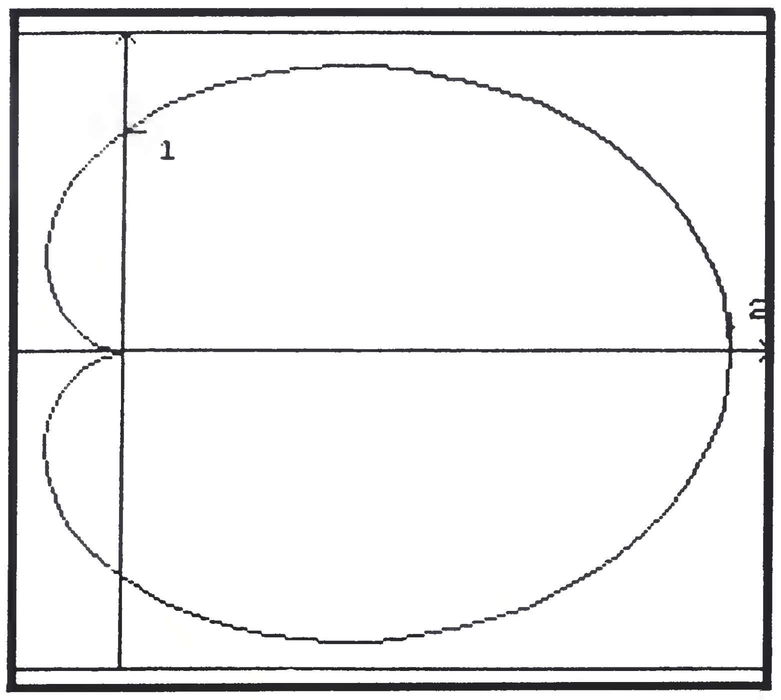
Rys. 3 Obwiednie krzywej



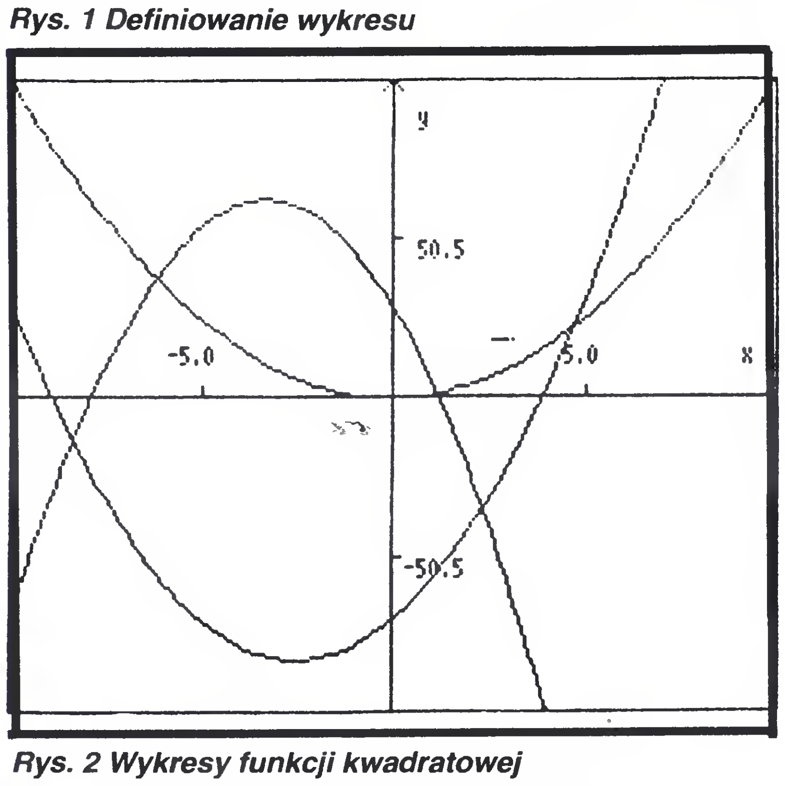
Rys. 4 Rodzina krzywych



Rys. 1 Definiowanie wykresu



Rys. 5 Kardioida



Rys. 2 Wykresy funkcji kwadratowej

producent:

VULCAN sp. z o.o.
51-657 Wrocław
ul. Kazimierska 11/1

5. Sposoby wykorzystania

Zdecydowanie najczęściej (w 98% przypadków) komputery są wykorzystywane do nauki o nich samych (zapoznanie się z komputerami, obsługa, podstawy programowania). Często stosuje je nauczyciel do demonstracji (69%) i niemal równie często pozwala korzystać swoim uczniom (67%).

6. Problemy i trudności

Wśród problemów i trudności związanych ze sprzętem najczęściej wymieniane są kłopoty z utrzymaniem go w stanie pełnej sprawności (66,4%). Trochę to dziwi, bo sporo szkół ma Juniory, do których producent powinien zapewnić serwis. Zbyt mało sprzętu ma 37% szkół, a nie wystarczającą ilość peryferii ponad połowa ankietowanych.

Kłopoty z oprogramowaniem koncentrują się na zbyt małej jego ilości (dokładnie 85% szkół wskazało na te trudności). Niemal równie dużą przeszkodą są niekompletne podręczniki i materiały pomocnicze (67%) i brak informacji o oprogramowaniu i jego jakości. Wielu nauczycieli narzeka na nie wystarczającą pomoc metodyczną i przyznaje się do niewielkiej wiedzy w zakresie obsługi i wykorzystania komputerów.

W sumie sytuacja nie przedstawia się w kolorowych barwach. Jest źle, a może być jeszcze gorzej. Będzie gorzej, jeśli decydenci szkolnictwa nie zajmą się problemem komputeryzacji na poważnie. Będzie gorzej, jeśli sami nauczyciele nie podniosą krzyku i nie zaczną walczyć o dostęp do sprzętu, szkolenia i oprogramowania.

Niezrozumiałe jest, dlaczego tak niewiele firm komputerowych nastawia się na współpracę ze szkołami. Być może nie jest to dostatecznie chłonny i bogaty rynek dla potentatów informatycznych sprzedających komputery setkami. Ale na pewno wiele osób znalazłoby godziwy zarobek choćby przy serwisie. Może po prostu za mało się o tym dotychczas mówiło?

T.B. Mańk

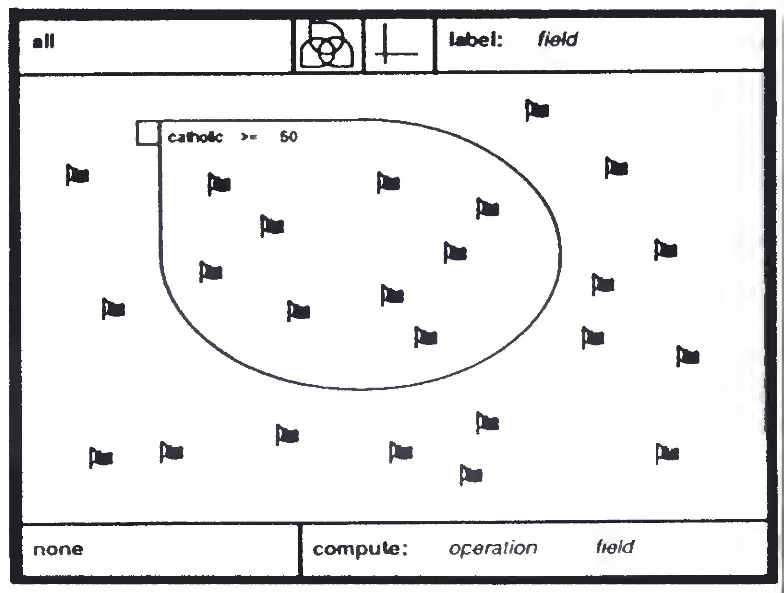
1) Badania przeprowadzono w Zakładzie Technologii Kształcenia Instytutu Pedagogiki Uniwersytetu im. A. Mickiewicza w Poznaniu w ramach resortowego programu RR I 16.

Praca została wykonana pod naukowym kierunkiem prof. dr. hab W. Strykowskiego.

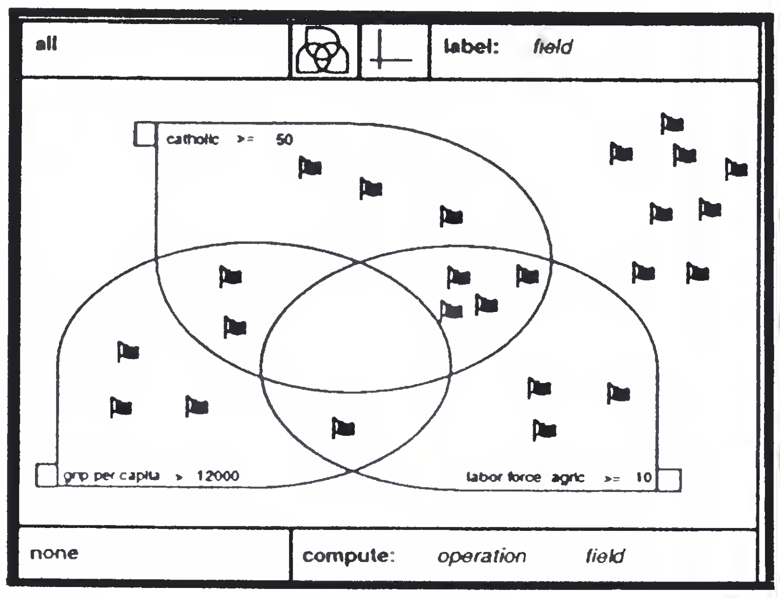
TABLE TOP

baza danych dla najmłodszych

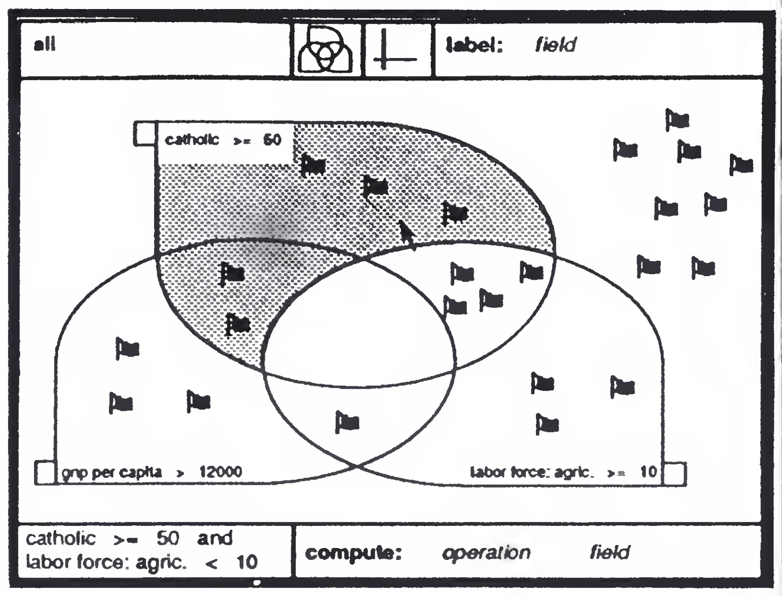
Zbliżający się XXI wiek będzie niewątpliwie wiekiem informacji. Zbieranie, sortowanie, wybór i korzystanie z różnego rodzaju danych będzie umiejętnością, bez której trudno wyobrazić sobie życie w cywilizowanym kraju. Umiejętność tę trzeba zdobywać już od najmłodszych lat, a najlepiej robić to używając stworzonych specjalnie do tego celu narzędzi: komputera i bazy danych.



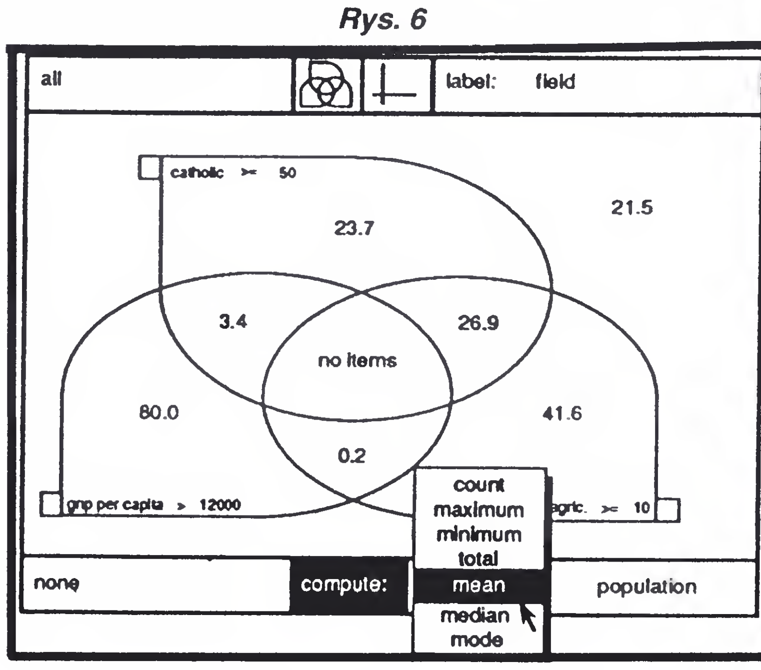
Rys. 4



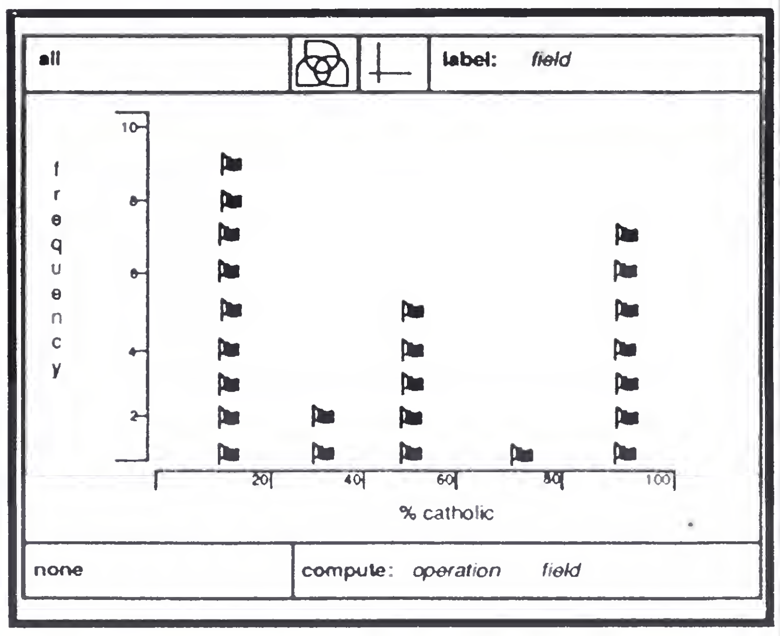
Rys. 5



Rys. 6



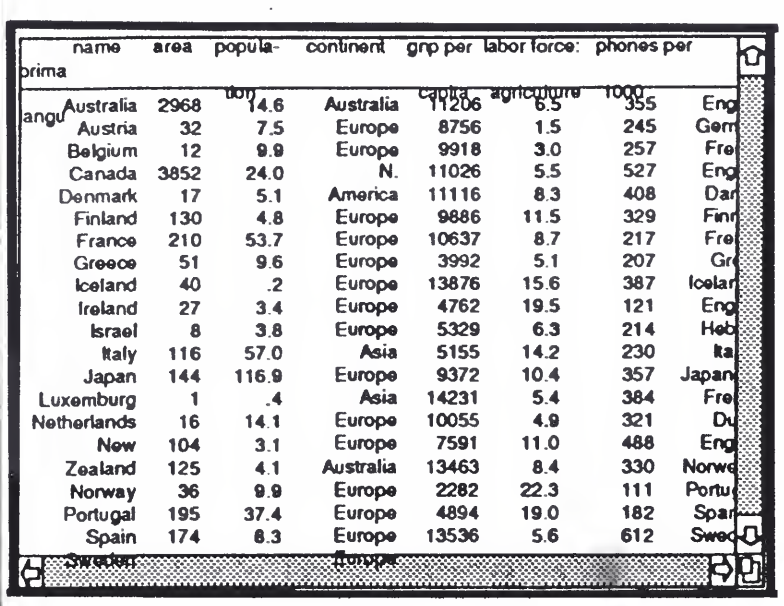
Rys. 7



Rys. 8

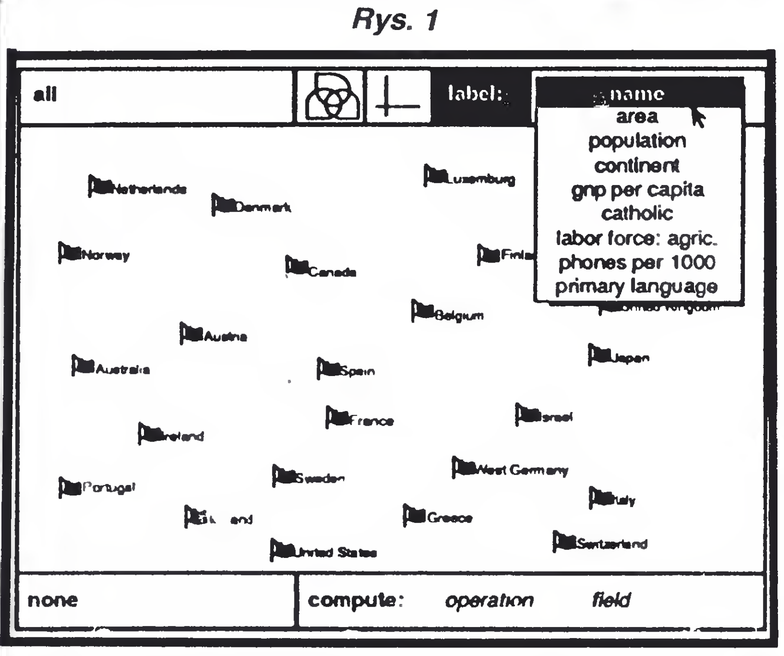
Baz danych używają poważni dorośli w poważnych i nudnych celach. Dzieci nie chcą używać programów, których jedynym efektem jest wypisanie kolumny nazw czy liczb. Nawet kiedy pojawiające się dane są dla dziecka interesujące, sposób ich prezentacji — zdecydowanie nie. Aby więc zachęcić dzieci do nauki, trzeba stworzyć nową efektowną formę posługiwania się informacją.

Tytułowy **TABLETOP** [(C) 1990 Technical Education Research Center (TERC)] jest właśnie próbą stworzenia łatwej w obsłudze i widowiskowej bazy danych. Podstawą tego programu jest baza danych typu arkuszowego: w kolejnych wierszach zgromadzone są dane dotyczące kolejnych obiektów. Przykładową bazę obejmującą 24 kraje europejskie

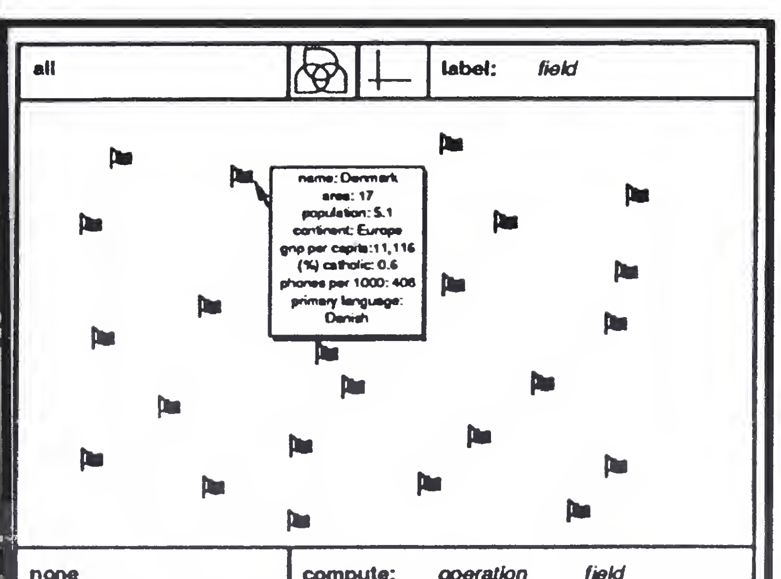


name	area	population	continent	gnp per capita	labor force: agriculture	phones per 1000	prima langu
Australia	2968	14.6	Australia	11206	6.5	355	Eng
Austria	32	7.5	Europe	8756	1.5	245	Gem
Belgium	12	9.9	Europe	9918	3.0	257	Fre
Canada	3852	24.0	N.	11026	5.5	527	Eng
Denmark	17	5.1	America	11116	8.3	408	Dan
Finland	130	4.8	Europe	9886	11.5	329	Finn
France	210	53.7	Europe	10637	8.7	217	Fre
Greece	51	9.6	Europe	3992	5.1	207	Gr
Iceland	40	.2	Europe	13876	15.6	387	Icelan
Ireland	27	3.4	Europe	4762	19.5	121	Eng
Israel	8	3.8	Europe	5329	6.3	214	Heb
Italy	116	57.0	Asia	5155	14.2	230	Ita
Japan	144	116.9	Europe	9372	10.4	357	Japan
Luxemburg	1	.4	Asia	14231	5.4	384	Fre
Netherlands	16	14.1	Europe	10055	4.9	321	Du
New	104	3.1	Europe	7591	11.0	488	Eng
Zealand	125	4.1	Australia	13463	8.4	330	Norwe
Norway	36	9.9	Europe	2282	22.3	111	Portu
Portugal	195	37.4	Europe	4894	19.0	182	Span
Spain	174	8.3	Europe	13536	5.6	612	Swed
Sweden			Europe				

Rys. 1



Rys. 2



Rys. 3

można obejrzeć na rys. 1. Nie jest to nic nowego; typowy arkusz elektroniczny. Taki był zresztą cel autorów: przygotować do pracy z „prawdziwymi" bazami danych. Taka prezentacja służy jednak tylko do wprowadzania danych.

Po wprowadzeniu danych każdy obiekt jest reprezentowany przez obrazek — ikonę (którą możemy sami zaprojektować). Przy każdej z nich może występować jej nazwa (rys. 2), o każdej możemy też uzyskać pełną informację, jak to widać na rys. 3.

Najciekawsza jednak część programu, a jednocześnie jego główna siła dydaktyczna leży w możliwości grupowania obiektów. Definiowanie interesującego nas podzbioru jest łatwe, a sposób prezentacji (rys. 4) jasny i zrozumiały. Gdy zdefiniujemy kilka podzbiorów, jak na rys. 5, to, niejako przy okazji, otrzymamy sporą ilość informacji z algebry zbiorów: części wspólne, zbiory rozłączne, suma zbiorów są pojęciami, które same się narzucają. Gdy do tego dodamy możliwość zaznaczenia podzbiorów spełniających wybrane równanie (rys. 6), to nauczyciele pierwszych klas mogą zacierać ręce. Przecież to jest właśnie to, co dotychczas usiłowali przedstawić na tablicy malując i wycierając setki ludzików, jabłek czy kółeczek, a potem obwodząc je liniami o fantastycznych kształtach.

TABLETOP jako pomoc przy nauce podstaw matematyki pozwala łatwo wykazać, **po co** w ogóle uczymy dzieci algebry zbiorów. Dlaczego jest to ważne i podstawowe zagadnienie matematyki, że od niego zaczyna się jej nauczanie. Dzieci też to widzą i dzięki temu łatwiej i chętniej się uczą.

Jednak ten program nie powstał z myślą o nauce algebry zbiorów. Pisząc o zbiorach chcieliśmy tylko wykazać, że dobre narzędzie można wykorzystać do wielu różnych celów. Używając **TABLETOP** można, jak w zwykłej bazie danych, wyliczać średnie, sumy, szukać maksymalnych i minimalnych wartości itp. (rys. 7). Dane możemy przedstawić w postaci tabelek, histogramów (rys. 8), wykresów i innych.

Program jest napisany dla komputera Macintosh II i wykorzystuje wszystkie jego zalety: obsługę myszką, komunikację z użytkownikiem za pomocą systemu ikon, okien i rozwijanych menu. W trakcie prób programu, jego twórcy pozwalali uczniom samodzielnie wybierać zakres, wprowadzać dane i poddawać je opracowaniu. Dzieci wykazywały dużą aktywność także przy wyborze tematu opracowania (uczniowie wybierali badanie form spędzania przerwy obiadowej).

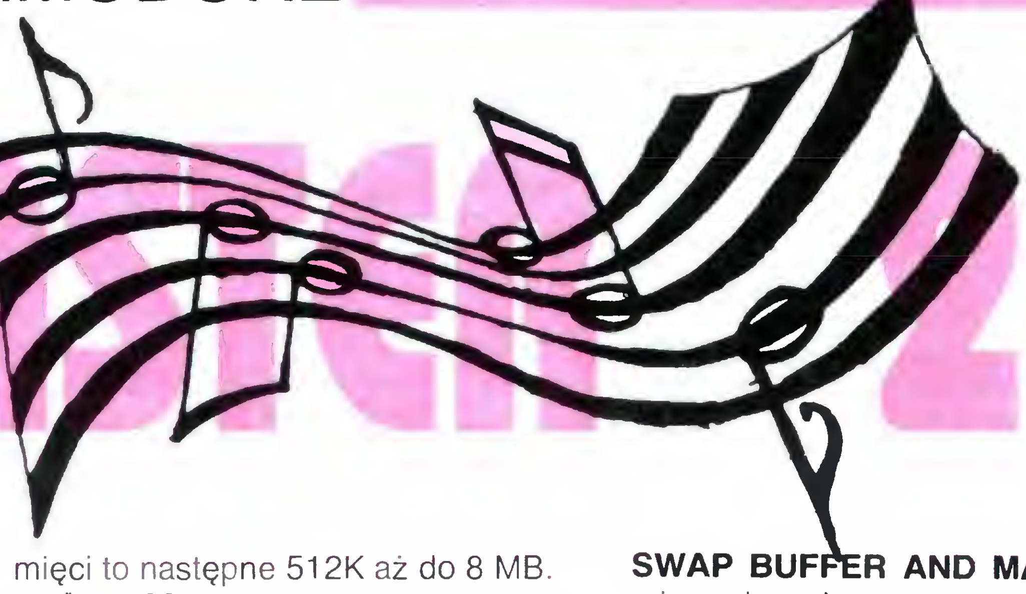
Jak większość programów uniwersalnych, **TABLETOP** można wykorzystać w wielu przypadkach, zależnie od potrzeb. Czy prezentowana na rysunkach baza danych zawierająca informacje o 24 państwach europejskich jest programem pomocnym w nauce geografii, algebry zbiorów czy informatyki? Wszystkie trzy możliwości są prawdziwe, którą wybierzemy — zależy tylko od nas.

Na koniec jeszcze jedna uwaga natury filozoficznej: dlaczego takich programów nie ma u nas? Polscy programiści z pewnością potrafiliby je napisać, a dydaktycy opracować koncepcję. Ale sprzedać można by tylko jedną kopię. Resztą zajmą się piraci. A ponieważ klienci kupują u nich, bo taniej, nauczyciele są skazani na oglądanie nowinek z zagranicy.

I.M.

1) (C) 1990 Technical Education Research Center (TERC)

AUDIOMASTER 2



Jednym z zastosowań Amigi jest dźwięk, a raczej duże możliwości jego przetwarzania z jednoczesnym zachowaniem dobrej jakości. Każdy jednak zgodzi się ze zdaniem, że niewiele można przetwarzać bez odpowiedniego programu.

Dla tych, którym nie wystarczają instrumenty dostępne w programach muzycznych, otwiera się „wielki świat przygody". W dowolny wręcz sposób możemy bawić się cyfrowym zapisem muzycznym, szukając tego brzmienia, które nas satysfakcjonuje. „Fachowość" nabyta tą drogą jest bardzo dobrym fundamentem do zrozumienia samej istoty dźwięku i genezy jego powstawania.

Audiomaster II jest już pewnym standardem programowym, jeżeli chodzi o obsługę digitalizowanego dźwięku. Wiele ciekawych operacji dostępnych dla użytkownika stawia ten program nad innymi, a szybkość działania decyduje o komforcie pracy.

Przyjrzyjmy się bliżej temu programowi.

Nasz ekran to dwie części, z których górna (robocza) przedstawia odwzorowanie graficzne danych w pamięci. Dolna partia (ekran funkcji) zawiera najczęściej stosowane funkcje programu, łatwo dostępne dla użytkownika. Patrząc od lewej są to:

SNAPSHOT — natychmiastowy zapis danych z pamięci na dyskietkę pod nazwą „snapshot".
RECALL — szybkie wczytanie z dyskietki pliku o nazwie „snapshot".
PLAY WAVEFORM — odtworzenie danych zawartych w pamięci.
PLAY DISPLAY — odtworzenie fragmentu danych zobrazowanych na ekranie roboczym (w danej chwili), z jednoczesnym wskazaniem pozycji liczonej w bajtach lub sekundach (patrz USER OPTIONS).
PLAY RANGE — odtworzenie wycinka danych wskazanych przez operatora w następujący sposób: ustawiamy wskaźnik na początku wybranego fragmentu, wciskamy lewy przycisk myszy i trzymając go przesuwamy na koniec interesującego nas obszaru.
STOP — natychmiastowe zatrzymanie odtwarzanych danych.
SHOW RANGE — graficzne powiększenie interesującego nas fragmentu.
SHOW ALL — wyświetlenie wszystkich danych zawartych w pamięci komputera.
ZOOM OUT — zwiększenie wyświetlanego obszaru danych dwukrotnie, aż do osiągnięcia pełnego obrazu dźwięku.
RANGE ALL — zaznaczenie całego wyświetlanego obszaru.
››› ‹‹‹ — precyzyjne przesuwanie wskaźników pętli.
SEEK ZERO — poszukiwanie wartości zerowych dla wskazanego typu danych .
SEEK LOOP — poszukiwanie powtarzającego się fragmentu danych.
LOOP ON (NO LOOP) — uruchomienie (wyłączenie) odtwarzania danych w pętli ograniczonej wskaźnikami.

Pozostałe funkcje zawarte są w następującym menu:

PROJECT:

LOAD TO EDIT WINDOW — wczytanie danych z dyskietki do obszaru ekranu roboczego.
LOAD TO COPY BUFFER — wczytanie danych z dyskietki do bufora.
LOAD TO RAM SCAN — $000000 wyświetlenie danych zawartych w pierwszych 512K pamięci RAM
— $200000 j.w., dostępny obszar pamięci to następne 512K aż do 8 MB.
— $C00000 obszar popularnego rozszerzenia pamięci do 1MB o symbolu A501 (i pochodnych).
SAVE WAVEFORM zapis na dyskietce całości danych z ekranu roboczego.
SAVE RANGED DATA — zapis na dyskietce danych wybranych przez użytkownika.

SAMPLER uruchomienie samplera (nagrywanie), po ustaleniu długości nagrywanych danych (w bajtach) i częstotliwości próbkowania; opcjonalnie możliwość oczekiwania na podanie sygnału.
MODE — przełącza tryb pracy programu z MONO na STEREO i odwrotnie (na życzenie z konwersją danych).

EDIT:

CUT.COPY.PASTE.CLEAR BUFFER<REPLACE — służą do operacji na danych takich jak: kopiowanie, przenoszenie, kasowanie zawartości bufora itp.
ZERO — zerowanie wskazanego obszaru (RANGE) lub bufora danych (COPY BUFFER).
INVERT — zmiana fazy zaznaczonego obszaru (o 180 stopni).
SWAP CHANEL — zamiana kanałów (tylko w opcji STEREO).
EDIT FREE HAND — coś dla cierpliwych — ręczne tworzenie wykresu.
CLEAR ALL DATA — skasowanie wszystkich danych w pamięci komputera.

EDIT 2:

REPLICATE LOOP — powtórzenie fragmentów danych zawartych pomiędzy wskaźnikami pętli (za końcowym wskaźnikiem w obszarze definiowanym przez funkcję ADD WORKSPACE).
ADD WORKSPACE — tworzy przerwę w danych o ustalonych w bajtach rozmiarach od miejsca ustalonego przez kursor.
SWAP BUFFER AND MAIN — wymiana danych zawartych w buforze z danymi ekranu roboczego.

SPECIAL FX:

ECHO — z opcjami częstotliwości, tempa opadania głośności, liczby powtórzeń.
BACKWARDS — zapisanie wyznaczonego obszaru danych od tyłu.
MIX WAVEFORMS — sumowanie danych z ekranu roboczego i bufora danych — wynik operacji zostaje umieszczony w miejscu wskazanym przez kursor.
CHANGE VOLUME — pozwala na zmianę głośności wybranego obszaru, na podstawie głośności początkowej i końcowej definiowanej z poziomu tej instrukcji.
TUNE WAVEFORM — pozwala zamienić wysokość brzmienia (dostrojenie), oktawę i w niewielkim stopniu jakość odtwarzania danych z ekranu.
LOW PASS FILTER — włącza filtr dolnoprzepustowy tłumiący wysokie tony.
REAL TIME ECHO — próbkowanie dźwięku z pogłosem.

USER OPTIONS:

Są to funkcje służące do ustawiania sposobu komunikacji programu z użytkownikiem (komunikaty, typy danych). Mamy możliwość: obsługi filtrów, definiowania pętli, doboru palety koloru i ustawienia urządzeń wejścia/wyjścia oraz konfiguracji samplera.

HIFI:

REMEMBER REPEAT POINTS — skraca zapis o powtarzające się fragmenty.
HIFI SAVE — zapis na dyskietce.
PLAY HIFI — odtworzenie danych zawartych w pamięci ze zwiększoną dynamiką.

Olaf Kostarczyk
Mateusz Krause

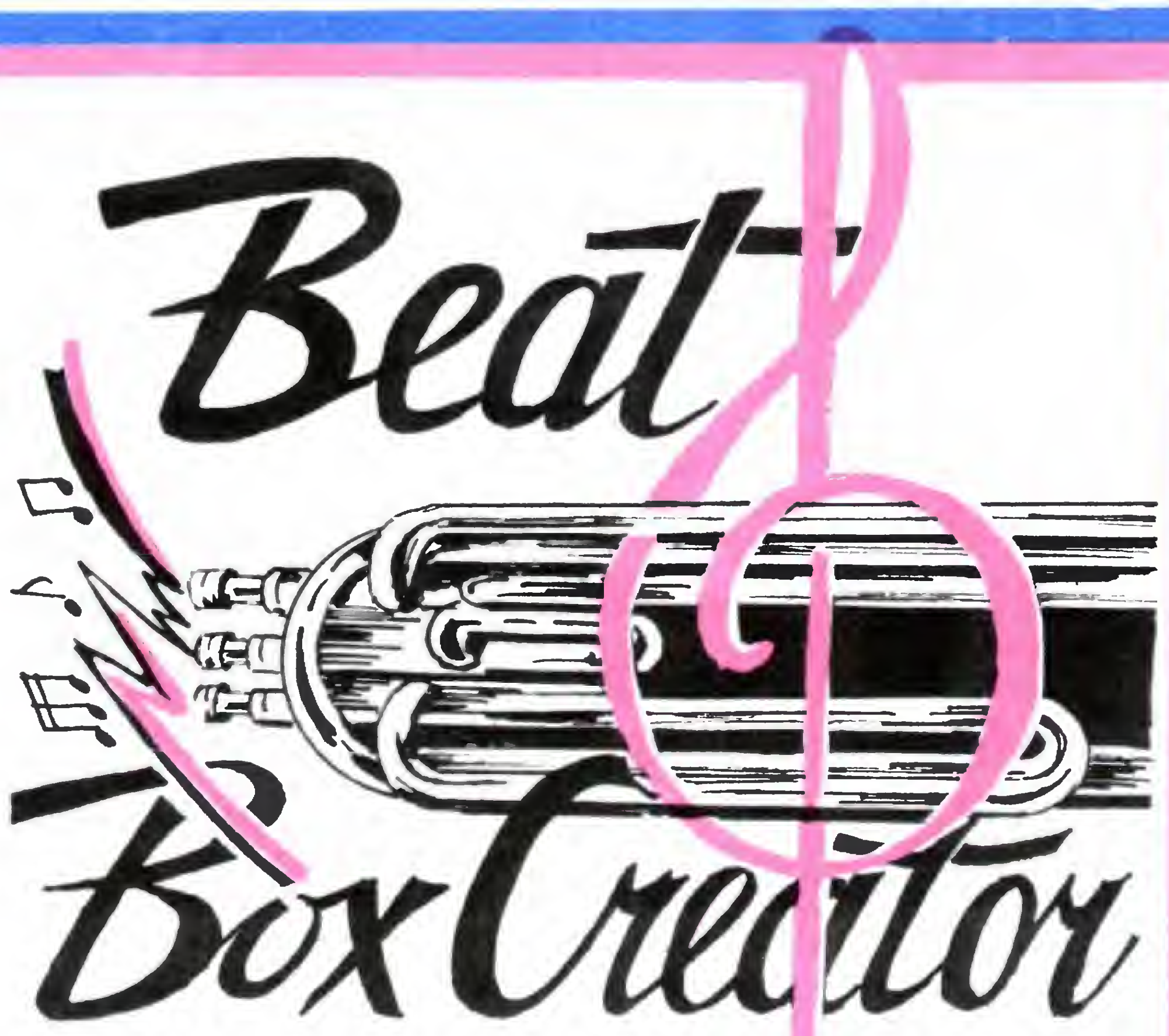
KODY REZYSTORÓW

Początkującym elektronikom wiele trudności sprawia nieraz odczytanie kolorowego kodu rezystora bez pomocy miernika lub katalogu.

Programik, który widać obok, przeznaczony jest do tłumaczenia kodu rezystorów na gotowe wartości liczbowe. Jego zaletą jest szybkość działania, a także to, że wynik drukowany jest od razu w kilku wielkościach tzn. w omach, kiloomach i megaomach. Przydatny on będzie w pracy, nie tylko początkującym, ale i poważnym elektronikom.

Piotr Liszewski

```
5 PRINTCHR$(147)
6 PRINT " ";TAB(5);"*** KOD BARW REZYSTOROW ***"
7 PRINT" ";TAB(40);"        CZARNY - 0 ","  ZIELONY - 5"
8 PRINT" ";TAB(40);"       BRAZOWY - 1 ","NIEBIESKI - 6"
9 PRINT" ";TAB(40);"     CZERWONY - 2 ","FIOLETOWY - 7 "
10 PRINT" ";TAB(40);" POMARANCZOWY - 3","     SZARY - 8"
11 PRINT" ";TAB(40);"          ZOLTY - 4","     BIA£Y - 9"
19 PRINT" ";TAB(40)
20 INPUT "    PODAJ KOD KOLORU 1 PIERSCIENIA";A
22 IF A=0 THEN PRINT" ";TAB(40);"  REZYSTANCJA REZYSTORA WYNOSI 0 OMOW":GOTO 20
23 IF A<0 THEN END
25 PRINT " ";TAB(40)
30 INPUT "    PODAJ KOD KOLORU 2 PIERSCIENIA";B
33 IF B<0 THEN END
35 PRINT" ";TAB(40)
40 INPUT "    PODAJ KOD KOLORU 3 PIERSCIENIA";C
50 LET Y=10*A+B
60 IF C=0 THEN PRINT" ";TAB(123);"WARTOSC REZYSTORA WYNOSI:";Y;"OMOW"
70 IF C=1 THEN PRINT" ";TAB(123);"WARTOSC REZYSTORA WYNOSI:";10*Y;"OMOW"
80 IF C=2 THEN PRINT" ";TAB(122);"WARTOSC REZYSTORA WYNOSI:";Y/10;"K OMOW"
90 IF C=3 THEN PRINT" ";TAB(122);"WARTOSC REZYSTORA WYNOSI:";Y;"K OMOW"
100 IF C=4 THEN PRINT" ";TAB(121);"WARTOSC REZYSTORA WYNOSI:";10*Y;"K OMOW"
110 IF C=5 THEN PRINT" ";TAB(122);"WARTOSC REZYSTORA WYNOSI:";Y/10;"M OMOW"
120 IF C=6 THEN PRINT" ";TAB(122);"WARTOSC REZYSTORA WYNOSI:";Y;"M OMOW"
130 IF C=7 THEN PRINT" ";TAB(122);"WARTOSC REZYSTORA WYNOSI:";10*Y;"M OMOW"
140 IF C=8 THEN PRINT" ";TAB(120);"BRAK W SZEREGU TAKIEJ WARTOSCI REZYSTORA"
150 IF C=8 THEN PRINT"     (REZYSTANCJA RZEDU";100*Y;"M OMOW)"
160 IF C=9 THEN PRINT" ";TAB(120);"BRAK W SZEREGU TAKIEJ WARTOSCI REZYSTORA"
170 IF C=9 THEN PRINT"     (REZYSTANCJA RZEDU";Y;"G OMOW)"
180 IF C<0 THEN END
200 FOR X=1 TO 5000
210 NEXT X
220 RUN
```



Jedną z wielu zalet Commodore'a 64 są wspaniałe możliwości muzyczne. Nie można sobie wyobrazić żadnej nowej gry bez atrakcyjnych wstawek muzycznych.

Wiele osób próbuje tworzyć własne utwory pisząc proste programy lub korzystając z gotowych narzędzi (MUSIC MASTER, MUSIC SHOP itp.). Są to programy gdzie stosuje się konwencjonalny zapis nutowy utworu. Dzięki nim można uzyskać jedynie mało atrakcyjne dźwięki przypominające raczej piski gumowych zabawek.

Stosunkowo młodą dziedziną w historii dźwięku na C64 jest łączenie zdigitalizowanych melodii (*sample*) ze standardowymi dźwiękami Commodore'a. Efektem jest utwór odbiegający niewiele od możliwości AMIGI. Pomocny w tworzeniu takiej muzyki jest program **BEAT BOX CREATOR**. W jego skład wchodzą: program główny, demonstracyjny oraz gotowe *sample* obejmujące dźwięki najważniejszych instrumentów. W programie znajduje się monitor dźwiękowy (symbole zapisywane są za pomocą kodu szesnastkowego), dzięki któremu tworzymy pewne sekwencje muzyczne łączone później z samplami.

Teraz trochę na temat samej obsługi programu. Po załadowaniu na ekranie pojawia się opis klawiszy funkcyjnych:

F1 — czyszczenie pamięci,
F2 — zakończenie programu,
F3 — ładowanie próbek (max 33 bloki),
F4 — edycja próbek,
F5 — testowanie próbek,
F6 — katalog dyskietki,
F7 — instalowanie próbki,
F8 — zapis utworu.

Klawisz **F1** powoduje skasowanie przestrzeni roboczej pamięci:
— od $ 4000 do $ 8FFF (+zainstalowanie próbki)
— od $ 9C00 do $ 9FFF (+ścieżki)

F2 kasuje program. Ponowny start przez SYS 4096 ($ 1000), po restarcie naciśnij **F1** aby wyczyścić pamięć.

F3 wywołuje procedurę ładowania próbek pod adres $ E000. Próbka nie może zajmować więcej miejsca na dysku niż 33 bloki.

Aby zrozumieć funkcję klawisza **F4** potrzebna jest odrobina wiedzy na temat zapisu szesnastkowego dźwięków w programie BEAT BOX CREATOR.

SN	STRT	STOP	SPD
FE	0000	0000	0090
FF	0000	0000	0090
00	0000	0000	0090
01	0000	0000	0090
02	0000	0000	0090

DD05

Oznaczenia
SN — sound number (numer dźwięku)
STRT — start address (adres początkowy)
STOP — stop address (adres końcowy)
SPD — speed value (szybkość)

Gdy zainstalujesz pierwszy dźwięk, musisz wybrać szybkość pierwszego *soundnumber-u*. Wartość pojawi się w linii *speedvalue*. Pozostałe wartości **BOX** dopisze sam. Możesz powielić dźwięk za pomocą klawiszy **F7** i **F8**. W tym:
— **F7** umożliwia ładowanie bieżącego dźwięku do bufora pamięci,
— **F8** umożliwia kopiowanie bufora do bieżącego dźwięku,
— **RETURN** — edycja próbki (szybkość, start i stop adres)
— **P** — przesłuchanie próbki (*play*).

F5 umożliwia testowanie ładowanych lub wykonanych prób k naciśnięciu SPACE (*keyrepeatsp ed*), następnie szybkość odegra a samej próbki (*samplespeed*, $ DD04-$ DD05) DD05)

RUN/STOP — zaniechanie kcji

F7 — umożliwia zainstalowanie próbki. Pytanie: czy jeste wny Y/N (Y — tak), zainstalować a maksymalnie 15 próbek.

SPEED: szybkość pró $ DD04 — $ DD05)

Naciśnięcie klawisza **F8** woduje nagranie na dysku **BOX**. W do stacji dyskietkę z głównym p ramem i naciśnij **SPACE**. W tej ch komputer załaduje **Soundmonit Editor**. Gdy skończy, włóż czystą ormatowaną dyskietkę i naciśnij **SPACE**. Następnie dokonaj kompres wojego utworu.

Możesz teraz posłuchać swojej muzyki w całości. Ocenę programu pozostawiam Tobie.

Piotr Lis wski
Waldemar owak

program: BEAT BOX EDITOR
autor: Klaus XIV
firma: nieznana

ZNAKI EKRANOWE COMMODORE

Dość często otrzymywałem listy z prośbą o wyjaśnienie, czy ukazywanie się na ekranie dziwnych znaczków (zwykle po otwarciu cudzysłowu) nie świadczy o uszkodzeniu komputera. Aby spełnić prośby szerokiego grona Czytelników, zamieszczam więc listę podstawowych znaków ekranowych i odpowiadających im klawiszy.

Znaki te są swoistym ułatwieniem, pozwalającym na oszczędne gospodarowanie pamięcią. Odnoszą się one nie tylko do oznaczania nadanych przez użytkownika kolorów, ale również spełniają takie funkcje, jak przesuw kursora w określone miejsce ekranu, lub skasowanie zawartości ekranu.

Znaki te pojawiają się na ekranie po otwarciu cudzysłowu; jeżeli jednak kogoś bardzo denerwują, to w zamian może uzyskać przypisane im funkcje poprzez zastosowanie odpowiednich kodów ASCII i funkcji CHR$() (wartości podano w instrukcji obsługi komputera). I tak np., aby przesunąć kursor w lewy górny róg ekranu (pozycja HOME), wystarczy wykonać:

PRINT CHR$(19)

Skasowanie zawartości ekranu odbywa się za pomocą:

PRINT CHR$(147)

Polecenie pierwsze odpowiada dokładnie wciśnięciu klawisza CLR/HOME, polecenie drugie natomiast — SHIFT CLR/HOME.

Jedynie od woli użytkownika i jego wygody zależy, którą metodę wybrać. Być może metoda druga jest nieco bardziej przejrzysta, ale zależy to chyba od pamięci wzrokowej.

Stosowanie funkcji CHR$ można uprościć przypisując odpowiedni kod ASCII zmiennej: Przypisz najczęściej używane znaki dowolnie wybranej zmiennej. Następnie wyświetl je za pomocą PRINT, tak jak to podano poniżej:

```
10 CL$ = CHR$(147)
20 DN$ = CHR$(17)
30 PRINT CL$; DN$; DN$; DN$
40 PRINT CL$DN$DN$DN$
```

Linie 30 i 40 są równoważne — stosowanie średnika nie jest tu konieczne. Zwróć jednak uwagę, że linia 40 jest znacznie mniej przejrzysta aniżeli linia 30. Wykonanie programu spowoduje najpierw skasowanie zawartości ekranu, a następnie przesunięcie kursora o trzy wiersze w dół.

Znaczenie słów angielskich użytych w opisie jest następujące:
RIGHT — w prawo
LEFT — w lewo
UP — do góry
DOWN — w dół

Klaudiusz Dybowski



CTRL 9
CTRL 0
C = 6
C = 7
C = 8
HOME
SHIFT CLR
SHIFT CRSR UP

CRSR DOWN
SHIFT CRSR LEFT
CRSR RIGHT
SHIFT INST
CTRL 1
CTRL 2
CTRL 3
CTRL 4
CTRL 5
CTRL 6
CTRL 7
CTRL 8
c = 1
c = 2
c = 3
c = 4
c = 5

1581 KŁOPOTÓW

Ze stacją Commodore 1581 przepracowałem ponad rok. Jeżeli miałbym ją opisać krótko, to powiedziałbym: „niezła jako druga stacja w zestawie. Jako jedyna — zdecydowanie odradzam".

1581 — to pierwsza stacja Commodore działająca **w oparciu** o dyskietki 3.5 cala. Z jednej strony jest to pewna wygoda, wynikająca jednak raczej z powodu przemyślanej konstrukcji samej dyskietki.

Stacja ta oferuje 3160 wolnych bloków na dyskietce oraz możliwość tworzeni· partycji (wydzielonych obszarów). Jest tu bardzo wygodne, gdy np. jedną dyskietkę chcemy wykorzystać do kilku różnych celów, ponieważ partycja taka pozwala na grupowanie określonych programów czy zbiorów w jednym miejscu. Partycje te jednak mają pewną kapitalną wprost wadę.

Biorąc pod uwagę, że obecna cena dyskietek 3.5 cala jest niewiele większa od dyskietek o średnicy 5.25, możemy nabrać mylnego przekonania, iż zakup tej stacji jest przykładem oszczędności. Podczas moich prób stwierdziłem, że:

1. 1581 jest rzeczywiście udana jako magazyn danych; jeśli ktoś zawodowo zajmuje się np. przetwarzaniem tekstów, tłumaczeniami czy grafiką komputerową, to stacja ta może spełnić swoje zadanie.
2. Stacja ta współpracuje bez problemów z systemem GEOS 128, co ma duże znaczenie dla zwolenników tego programu; duża pojemność dyskietki (ok. 760 KB) ułatwia pracę. Brak natomiast możliwości współpracy z wersją GEOS 64 (przynajmniej w wersjach od 1.0 do 1.4). Jednak nawet i tu są pewne ograniczenia.
3. Współpraca z Commodore 128 odbywa się w trybie przyśpieszonym, co skraca czas oczekiwania na wczytanie programu lub zbioru do pamięci. Udogodnienie to nie istnieje w wypadku współpracy z C-64.
4. Specjalny tryb pracy (BURST) pozwala na rzeczywiście błyskawiczne operacje na zbiorach; problem polega na tym, że tryb ten jest dostępny z poziomu języka maszynowego i wcale nie jest tak łatwo (nawet dla zaawansowanych w tej dziedzinie) ułożyć odpowiedni program.
5. Różnica szybkości pracy jest spowodowana dwoma przyczynami. Po pierwsze, zwiększony bufor pamięci RAM pozwala na wczytanie doń zawartości całej ścieżki. Po drugie, stacja odpowiada na wysyłany przez Commodore 128 sygnał Host Request Fast. Siłą rzeczy, C-64 nie jest w stanie takiego sygnału wytworzyć.

Jak wiadomo, stacja 1581 pracuje ze zmienionym formatem zapisu (MFM zamiast GCR). Dyskietka jest podzielona na 80 ścieżek, po 40 sektorów na ścieżce. Istnieje różnica pomiędzy interpretacją DOS i stanem fizycznym struktury dyskietki; ma to, jak się okazuje, swoje złe strony ujawniające się w całej okazałości podczas wykonywania operacji VALIDATE. Kilkakrotnie, z zupełnie dla mnie niewytłumaczalnych przyczyn, stacja zgłaszała błąd w odczycie dyskietki, zwykle pod koniec VALIDATE; przy próbach odczytu okazywało się, że uszkodzona jest ścieżka 40, na której 1581 umieszcza katalog. Znamienne jest, że we wszystkich wypadkach skasowaniu (READ ERROR) ulegało zawsze pierwszych 20 sektorów danej ścieżki.

Sytuacja ta jest nad wyraz nieprzyjemna, gdy w stacji znajduje się dyskietka systemowa, o której odtworzeniu niedoświadczony użytkownik nie ma co marzyć. W moim przypadku odtwarzanie zajęło ponad trzy dni i nie wszystkie programy udało mi się naprawić. Dlatego też napisałem, że oszczędność wydaje się tu tylko teoretyczna — praktycznie użytkownik powinien mieć na wszelki wypadek zapasową kopię dyskietek z danymi, a na pewno co najmniej 3 dyskietki systemowe. Aby zabezpieczyć się dodatkowo, na wszelki wypadek przeniosłem wszystkie programy na dyskietkę 5.25 cala. W wielu wypadkach musiałem kopiować dane na inną dyskietkę przed próbą wykonania VALIDATE. Szczerze mówiąc, jest to dość uciążliwe. Przy pracy z 1581 naprawdę warto trzymać swoje dyskietki zabezpieczone przed zapisem.

Stacja 1581 nie współpracuje z FINAL CARTRIDGE ani z podobnymi kartami czy programami przyśpieszającymi. Ponadto, ze względu na zmieniony format, nie pasują do niej żadne programy kopiujące. Oczywiście można skorzystać tu z programu FILE-COPY czy UNICOPY z dyskietki systemowej pod warunkiem, że użytkownik ma sporo czasu.

Duża pojemność dyskietki pozwala na zapisanie maksymalnie 296 plików. Niestety jest to nierealne, gdy stacja pracuje pod kontrolą systemu GEOS — maksymalna dopuszczalna liczba plików wynosi wówczas 143.

Wspominałem już wcześniej, że partycje (rozszerzenie CBM w miejscu PRG) mają pewną wadę. Partycja w rozumieniu CBM fizycznie wygląda jak rzeczywista partycja dysku twardego zarządzanego poprzez DOS, natomiast w rzeczywistości pełni ona rolę podkatalogu. Po utworzeniu partycji (musi mieć co najmniej 120 bloków i zawiera własną ścieżkę systemową z katalogiem) należy przejść do niej i sformatować ją. Wada, o której chcę powiedzieć, umożliwia SKASOWANIE PARTYCJI bez względu na to, czy zawiera ona jakieś programy, czy nie (w MS-DOS skasowanie podkatalogu jest niemożliwe, jeśli zawiera on program). Innymi słowy: chcesz stracić rezultaty wielogodzinnej pracy — wpisz je do partycji.

Stacja ta jest bardzo słabo oprogramowana. Przypuszczam bowiem, że i producenci oprogramowania wzięli ją na warsztat wykrywając te same wady; z urządzenia tego można więc korzystać jedynie w standardowych zastosowaniach. Jeśli chodzi o mniejsze firmy, to spotkałem kilka ogłoszeń w prasie amerykańskiej na temat oprogramowania (przede wszystkim kopiującego) dla tej stacji. Ukazało się też kilka programów użytkowych przeznaczonych właśnie dla 1581, które jak dobrze pójdzie, niebawem Czytelnikom przedstawimy.

I jeszcze jedna informacja. Być może jest to plotka, a być może nie. Od kilku osób słyszałem, że możliwe jest przerobienie stacji 1581 na dodatkową stację do AMIGI. Ile jednak w tym prawdy i jaki jest koszt przeróbki — nie wiem.

Zwykle w tym miejscu powinienem wyciągnąć wnioski. Tym razem zostawiam to Czytelnikom; jedyne, co mi się nasuwa, to smutek, że całkiem dobry pomysł po raz kolejny firma Commodore zawaliła.

Klaudiusz Dybowski

```
0 [illegible]
760   "PAL"          CBM<
760   "MADS"         CBM<
760   "MERLIN"       CBM<
760   "LADS"         CBM<
120 BLOCKS FREE.

READY.
```

Zabezpieczone przed usunięciem (znaczek „<") partycje na dyskietce 3,5 cala.

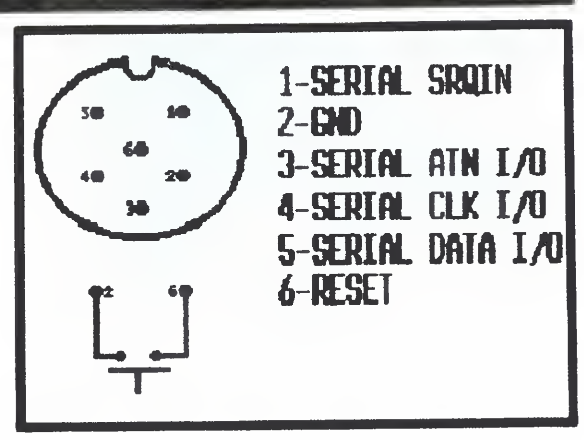
RESET STACJI DYSKÓW

Stacja dysków jest dość delikatnym urządzeniem. Ciągłe włączanie i wyłączanie jej gdy się „zawiesi" na pewno nie wpływa dodatnio na jej żywotność, zaś wypisywanie ciągu komend inicjalizujących stację nie zawsze daje rezultat.

Proponuję Ci samodzielne wykonanie przycisku RESET w stacji dysków. Usprawnienie to może jednak zostać wykorzystane tylko wtedy, gdy stacja nie resetuje się po skasowaniu pamięci komputera za pomocą modułu (np. FINAL III). W innych przypadkach używanie RESET może być niebezpieczne.

Wykonanie przycisku jest zupełnie niewskazane w przypadku C128D, gdyż posiadają one RESET wbudowany fabrycznie.

Istota rozwiązania to połączenie pinów 2 i 6 gniazda SERIAL stacji dysków za pomocą mikroprzełącznika. Po wciśnięciu mikroprzełącznika stacja powinna się zresetować nie kasując pamięci C64.



Przycisk RESET na pewno ułatwi Ci pracę, a Twoja stacja nie będzie nękana ciągłym wyłączaniem i włączaniem.

Piotr Liszewski

AST – odpowiedź

W „BAJTKU" nr 4 (11–12)/89 ukazał się artykuł Jakuba Cebuli zawierający parę słów krytycznych pod adresem systemu AST. Firmie AS wypada tylko podziękować autorowi za zdopingowanie nas do zapoznania Państwa z aktualnym oprogramowaniem do systemu ATARI SUPER TURBO, jesteśmy to winni naszym obecnym i przyszłym klientom. Jak napisał autor wspomnianego artykułu, opisane zostało oprogramowanie z połowy 1989 roku, a obecnie mamy rok 1991, półtora roku w informatyce to epoka.

System AST powstał całkowicie w Polsce, jako pierwszy system **turbo**, zarówno oprogramowanie jak i przeróbka sprzętowa nie były na niczym wzorowane. Głównie z tego powodu pierwsze oprogramowanie do systemu zawierało wiele niedoskonałości i błędów, które staraliśmy się w miarę upływu czasu usuwać. Większość osób projektujących nowe systemy turbo opierała się na naszych doświadczeniach, inni przerabiali nasze programy (głównie napisy) i nie martwiąc się o nic sprzedawali jako własne. Wyliczanie poszczególnych etapów ewolucji systemu nie miałoby sensu. Chcielibyśmy jednak zwrócić uwagę, że każda zmiana wiązała się z zasadą, że wszyscy nasi „byli" klienci muszą mieć możliwość korzystania ze wszystkich nowo powstających programów, jako punkt honoru postawiliśmy sobie posiadanie wszystkich programów w systemie AST (co nam się udało, jest tylko kilka programów nie pracujących w tym systemie, ale jest też sporo programów, które współpracują tylko z dyskietką i AST). Dodatkowy przewód, tak charakterystyczny dla systemu, jest właśnie pozostałością po pierwszej wersji dla magnetofonu 1010, jest więc tylko wynikiem chęci utrzymania standardu. Choć obecnie oferowane oprogramowanie nie ma już wiele wspólnego z początkowym jednak nawet nasi pierwsi klienci nie mają problemu z korzystaniem z niego. Ponieważ uwagi we wspomnianym na wstępie artykule dotyczą starego oprogramowania, więc z większością się zgadzamy. Poniżej postaramy się jednak wyjaśnić wątpliwości i problemy autora.

Pierwszy dotyczył braku obsługi urządzenia T:. Istnieje już od dość dawna program EMULATOR AST. Powstał on właśnie do kopiowania programów użytkowych i automatycznie instaluje procedurę obsługi Turbo. Przy okazji posiada jeszcze kilka zalet, ma możliwość kopiowania programów współpracujących z dyskietką pod DOS-em na system AST, umożliwia kopiowanie programów wieloblokowych z pierwszym blokiem w formacie DOS-u lub Basica, instaluje w kopiowanych programach bufor 4KB (znika pojęcie długich czy krótkich przerw między rekordami) itd. Program znajduje się w obszarze DOS-u, czyli tym, z którego nie korzysta większość programów użytkowych, więc świetnie nadaje się do ich kopiowania.

Druga uwaga dotyczyła podstawowych kopierów, zgadzamy się w pełni z niedoskonałościami tych programów, mogą być one dość uciążliwe. Obecnie proponujemy Państwu inne kopiery, o których napiszemy poniżej. Również zgadzamy się z zastrzeżeniami dotyczącymi AST Basica, program zawierał rzeczywiście sporo błędów (jednak jako kopier z Normal na AST, dla programów napisanych w Basicu działał prawidłowo). Wszystkie jego niedoskonałości wyeliminował EMULATOR. Przy jego użyciu można pracować w Basicu poprawnie i bez kłopotów, a praca w tym formacie przynosi wiele nowych korzyści.

Parę słów należy poświęcić Turbo Basicowi XL. Program, który Państwo otrzymywali gratis do pakietu specjalnego, także nie był doskonały i głównie mógł służyć jako „zbiór" do uruchamiania programów napisanych w tym języku. Potem proponowaliśmy naszym klientom program napisany nie dla nas, powstał on w Łodzi, nie znamy niestety autora („piractwo" należy tłumaczyć brakiem czasu na stworzenie przyzwoitej wersji, wieloma prośbami klientów i usprawiedliwić, że wszystkie programy tworzyliśmy sami, wiele zaś firm sprzedawało je bez zezwolenia), ...ale rzeczywistość okazała się nieubłagana, niedoskonałość także tego programu zmusiła nas do napisania programu działającego już w pełni poprawnie, powstał on także na bazie standardu EMULATORa.

Ostatnie uwagi dotyczą Pakietu Specjalnego. Dziwi nas bardzo, że p. Cebuli nie udało się wykorzystać jego możliwości; podstawowym programem pakietu jest CHANGER, który w naszej firmie jest używany bardzo często. Można przy jego użyciu przekopiować większość programów wieloblokowych oraz prawie wszystkie programy bootingowe. Instrukcja może zawierać trochę zbyt dużo informacji, ale miała ona służyć zarówno osobom zaawansowanym, jak i początkującym.

W podsumowaniu wspomnianego na wstępie artykułu znajdujemy sugestie na temat unowocześnienia rozwiązań sprzętowych i montowanego w Krakowie Universal Turbo. To chyba mała pomyłka, układ montowany do magnetofonów AST jest praktycznie pierwowzorem większości instalowanych w naszym kraju systemów i nie ma on żadnych problemów z pracą w większości z nich (wystarczy jedynie zamontować we wtyczce przełącznik podłączający do masy dodawany przez nas dodatkowy przewód, np. dla Turbo 2000, ewentualnie trochę inaczej podłączyć płytkę, co można także dokonać przy użyciu przełącznika), mamy wielu klientów, którzy korzystają z kilku systemów posiadając magnetofon w AST, czyli równie dobrze nasz układ można nazwać uniwersal turbo (to tylko kwestia nazwy). Natomiast jest nam bardzo przykro, że jakaś firma rozprowadza nasze oprogramowanie bez naszej zgody (prawdopodobnie Blizzarda także), jeżeli chcemy uporządkować ten rynek, pewne zasady powinny istnieć. Wróćmy jednak do sprawy uniwersalności modułu turbo. Ponieważ jesteśmy prawnym dystrybutorem systemu AST, więc w Państwa magnetofonach instalujemy ten system i na działanie w tym systemie otrzymują Państwo gwarancję (oczywiście także w Normal), natomiast wszelkie próby pracy w innych systemach są Państwa indywidualną sprawą (nie jesteśmy w stanie gwarantować pracy w innych, często nam nie znanych systemach). Za system AST jesteśmy w pełni odpowiedzialni i opiekujemy się sprzętem naszych klientów w czasie gwarancji jak i po. Również od strony programowej mogą Państwo liczyć na naszą pomoc, a w systemie AST posiadamy praktycznie wszystkie programy.

Na zakończenie kilka słów o nowych programach, które wyeliminowały błędy wcześniejszych.

AST MONITOR — pracuje w AST, wolny bufor 56kB, posiada wbudowany assembler i disasembler, standardowe opcje monitora, bardzo wygodny kopier AST–AST, umożliwia kopiowanie programów, pisanie własnych, dokonywanie zmian itp.

AST TESTER — program do sprawdzania poprawności ustawienia głowicy i prędkości obrotowej silnika, znacznie dokładniejszy i zupełnie inny niż w pierwszej wersji systemu.

AST MULTICARTRIDGE — nowy, zupełnie inny cartridge do systemu (32kB lub w rozszerzonej wersji 64kB pamięci), wiele programów do obsługi systemu, m.in. wspomniane Emulator, Monitor, Tester, Unichanger itd., a także nowy standard MACRO COPY, pozwala na kopiowanie dowolnych zbiorów typu file o długości do 56kB we wszystkie strony pomiędzy AST i Normal, kopiuje prawie wszystkie programy z Normal na AST i odwrotnie, jest to nowy standard, znajduje się tu także jeden uniwersalny Multiloader, automatycznie wybierający jeden z trzech potrzebnych loaderów.

Ostatnią ciekawostką jest niewątpliwie seria modułów opartych na wcześniej wspomnianym. Proponujemy Państwu dowolne zestawy programów na cartridgeach robionych na zamówienie.

Tomasz Mazur
ATARI STUDIO „AS"

TAJEMNICE NIEŚMIERTELNYCH

DARG — zamieniamy: 4C, 28, 30 na 5C, 28, 30 (dziesiętnie: 76, 40, 48 na 92, 40, 48).
JAWA JIM — zamieniamy: D6, 01 na A5 01 (dziesiętnie: 214, 1 na 165, 1) oraz: D6, FC na A5 FC (dziesiętnie: 214, 252 na 165, 252).
NIGHT RIDERS — zamieniamy: C6, 9E na A5, 9E (dziesiętnie: 198, 158 na 165, 158).
ANGLE WORMS II — graczowi numer 1 pomagamy poprawką: E9, 01, 8D, EA na 54, 01, 8D EA (dziesiętnie: 233, 1, 141, 234 na 84, 1, 141, 234). Graczowi numer 2: E9, 01, 8D, EB na 54, 01, 8D, EB (dziesiętnie: 233, 1, 141, 235 na 84, 1, 141, 235). Teraz już obaj mają nieskończoną ilość amunicji.
THRESHOLD — zamieniamy: CE, 8F, 77 na AD, 8F, 77 (dziesiętnie: 206, 143, 119 na 173, 143, 119).
HENRI — zamieniamy: C6, C7 na A5, C7 (dziesiętnie: 198, 199 na 165, 199).
CREEP — zamieniamy: CE, OE, 4C na AD, OE, 4C (dziesiętnie: 206, 14, 76 na 173, 14, 76).
SCOOTER — Graczowi nr 1 pomagamy odnajdując dwa razy sekwencję: C6, A6 i zamieniając ją na: A5, A6 (dziesiętnie: 198, 166 na 165, 166). Graczowi nr 2 wystarczy jednokrotna zamiana: C6, A7 na A5, A7 (dziesiętnie: 198, 167 na 165, 167).
BATTY BUILDERS — zamieniamy: C6, 8B na A5, 8B (dziesiętnie: 198, 139 na 165, 139).
SCREMING WINGS — nieskończoną liczbę żyć zapewni poprawka: CE, 2E, 06 na AD, 2E, 06 (dziesiętnie: 206, 46, 6 na 173, 46, 6). Możliwość ratowania sobie życia „fikołkami" niezliczoną ilość razy umożliwi zamiana: CE, A5, 9F na AD, A5, 9F (dziesiętnie: 206, 165, 159 na 173, 165, 159).
EXCELSOR — zamieniamy: CE, E4, 19 na AD, E4, 19 (dziesiętnie: 206, 228, 25 na 173, 228, 25).
LEGGIT — zamieniamy: C6, D9 na A5, D9 (dziesiętnie: 198, 217 na 165, 217) oraz C6, DA na A5, DA (dziesiętnie: 198, 218 na 165, 218).
STORM — Wieczną energię zapewni zamiana: AD, BB, 85, 38 na 60, BB, 85, 38 (dziesiętnie: 173, 187, 133, 56 na 96, 187, 133, 56).
SPELLBOUND — dwa razy w programie zamieniamy: CE, FD, 03 na AD, FD, 03 (dziesiętnie: 206, 253, 3 na 173, 253, 3) i nasz bohater ma wieczne siły. Wyłączenie zegarów limitujących czas wszystkim uczestnikom gry następuje po wprowadzeniu dwóch kolejnych zmian: 3D, 60, A0 na 3D, 60, 60 (dziesiętnie: 61, 96, 160 na 61, 96, 96) oraz 1C, 60, 86 na 1C, 60, 60 (dziesiętnie: 28, 96, 134 na 28, 96, 96).

Tomasz Wiśniewski

KUPUJEMY Atari

Po długim oszczędzaniu i wielu wyrzeczeniach podjąłeś decyzję o zakupieniu komputera Atari. W tym momencie pojawiają się pytania: co, gdzie i jak kupić? Postaram się udzielić wskazówek, które pomogą odpowiedzieć na te pytania, ale ostateczną decyzję każdy musi podjąć samodzielnie.

CO?

Pierwszym problemem jest wybór rodzaju komputera i urządzeń zewnętrznych. Zacznijmy od komputera. Może to być 800XL, 65E lub 130XE. Ponieważ 800XL i 65XE różnią się w zasadzie tylko obudową, to ograniczmy się do wyboru pomiędzy 65 i 130. Były one już opisywane w artykule „Co jest co w Atari" („Bajtek" 2/91). Przypomnę jedynie, że różnią się one pojemnością pamięci operacyjnej — 65XE ma 64 KB RAM, a 130XE ma 128 KB.

Trzeba się więc zastanowić, czy potrzebna jest dodatkowa pamięć i jak będzie wykorzystana. Niemal wszystkie gry są pisane dla komputerów z minimalną pamięcią, więc w tym przypadku wybór jest zupełnie dowolny. Są oczywiście wersje gier przeznaczone dla 130XE, ale takie same gry istnieją również dla 65XE. Na przykład „Silent Service" wymaga odczytywania podczas gry niektórych danych z dyskietki, w wersji zaś na 130XE dane te są odczytywane przy uruchamianiu programu i przechowywane w dodatkowej pamięci.

Nieco inaczej przedstawia się to w przypadku programów użytkowych. Wiele z nich automatycznie rozpoznaje wielkość dostępnej pamięci i w zależności od tego pozwala na umieszczenie mniejszej lub większej liczby danych (informacji). Dotyczy to przede wszystkim arkuszy kalkulacyjnych, edytorów tekstu i baz danych. Są również programy pisane specjalnie w dwóch wersjach, które różnią się liczbą dostępnych funkcji. Jako przykłady można tu wymienić „SynCalc" i „Typesetter". Programy, które są pisane specjalnie dla 130XE i nie działają na 65XE, są stosunkowo nieliczne. Może się jednak okazać, że akurat taki program jest nam niezbędnie potrzebny. Dodatkową pamięć można jeszcze wykorzystać jako ramdysk, czyli dodatkową, bardzo szybką stację — jest to najczęstsze zastosowanie.

Wynikałoby z powyższych rozważań, że należy kupować tylko 130XE. Nie jest to prawda. Potrzeba posiadania większej pojemności pamięci operacyjnej pojawia się zwykle dopiero po pewnym czasie użytkowania komputera, a w dodatku dotyczy to także 130XE. Po prostu — zgodnie ze znaną w informatyce zasadą — każdy komputer ma za mało pamięci. Na szczęście liczne firmy wykonują rozszerzenia pamięci Atari do 192 lub 256 KB. Różnica w cenie takiej usługi jest znacznie mniejsza niż różnica cen 65XE i 130XE.

Sam komputer to jeszcze nie wszystko — niezbędna jest jeszcze pamięć masowa, czyli magnetofon lub stacja dysków. W tym przypadku wybór jest często zdeterminowany zasobnością kieszeni nabywcy. Za jedną stację można bowiem zakupić pięć magnetofonów. Jednakże stacja jest lepsza od magnetofonu znacznie więcej niż pięć razy. Trzeba bowiem wiedzieć, że początkowo konstruktorzy Atari w ogóle nie przewidywali możliwości korzystania z magnetofonu. Przede wszystkim stacja jest 32 razy szybsza. Korzystając z magnetofonu trzeba większość czynności przygotowawczych wykonywać osobiście (przewijanie taśmy, wyszukiwanie programu, włączanie zapisu i odczytu). Stacja dysków robi to wszystko sama, trzeba jedynie włożyć do niej dyskietkę. Ponadto przy odczycie z kasety bardzo często występują błędy, które powodują konieczność powtórnego odczytu. Przy korzystaniu ze stacji takie przypadki są bardzo rzadkie.

Niebagatelnym argumentem jest dostępność oprogramowania i możliwość jego właściwego wykorzystania. Niemal wszystkie programy są przeznaczone dla stacji dysków. Dopiero w Polsce powstają ich wersje kasetowe. Ich autorami są zwykle złodzieje (zwani eufemistycznie „piratami"), którzy często podczas wykonywania przeróbek uszkadzają programy. Nie jest to istotna przeszkoda, jeśli zamierzamy wykorzystywać komputer tylko jako zabawkę. Gdy komputer ma służyć do pracy lub nauki, stacja dysków jest urządzeniem niezbędnym. PAMIĘTAJ: komputer z magnetofonem utrudnia pracę, zamiast ułatwiać!

W moich kontaktach z rodzicami kupującymi komputer dzieciom często spotykam się z opinią: „Kupię magnetofon, a jak się dziecku spodoba, to dokupię stację". Jeżeli rzeczywiście dziecku się spodoba, to mamy wyrzucone pieniądze na magentofon (500 tys.), kasety (10 tys. za sztukę) oraz programy. W szczególnych przypadkach te zbędne wydatki mogą przekroczyć milion złotych. Jeśli natomiast dziecku się nie spodoba, to przede wszystkim oswoi się ono z „normalnym" sprzętem, a ponadto można komputer i stację sprzedać — zwykle łatwiej niż komputer z magnetofonem. Problem ten ma jeszcze jeden aspekt, który jest rzadko brany pod uwagę. Niecierpliwe, lecz zdolne dziecko często zniechęca się do komputera z powodu kłopotów z magnetofonem, co nie miałoby miejsca, gdyby od razu kupić mu stację.

Niekiedy myślimy jeszcze o zakupie drukarki. Tu powtórzę tylko to, co już często pisałem: drukarka może być dowolna pod warunkiem, że nie jest to Atari 1025, 1027 lub 1029. Mają się one do innych drukarek tak, jak Syrena do Forda lub nawet Mercedesa. Oczywiście wszystkie inne drukarki wymagają zastosowania interfejsu Centronics.

GDZIE?

Istnieją cztery możliwości — komputer można kupić w Pewexie, w komisie, na giełdzie i za granicą. Najdrożej i najlepiej jest w Pewexie. Otrzymujemy tam sprzęt nowy z gwarancją na cały rok. Kupowanie w komisie lub na giełdzie zawsze wiąże się z ryzykiem nabycia sprzętu uszkodzonego. Nie pomoże nawet najdokładniejsze sprawdzanie. Miałem kiedyś do czynienia z komputerem, który po 30–40 minutach zawieszał się, gdyż miał uszkodzony procesor. Konia z rzędem temu, kto wykryje taką usterkę przy zakupie.

Często praktykowane jest kupowanie za granicą. Otrzymujemy w ten sposób sprzęt nowy, przeważnie z gwarancją i taniej niż w Pewexie. Ale oddanie komputera do naprawy gwarancyjnej wymaga wyjazdu za granicę — trzeba więc mieć w rodzinie lub wśród znajomych kogoś, kto często jeździ na Zachód.

Używałem tu wszędzie słowa „komputer". Uwagi te dotyczą jednak wszystkich elementów systemu, czyli również magnetofonu i stacji dysków. Problem ten najwyraźniej uwidacznia się jednak w przypadku magnetofonu, gdyż często zachodzi konieczność regulowania jego głowicy.

JAK?

Odpowiedź na to pytanie zależy głównie od miejsca zakupu. Kupując sprzęt w Pewexie zabieramy go po prostu do domu i tam spokojnie sprawdzamy jego działanie. W sklepie nie ma na to warunków, a ewentualne usterki (które są mało prawdopodobne) usunie serwis.

Przy kupowaniu na giełdzie lub w komisie trzeba dokładnie sprawdzić sprzęt na miejscu. Niestety nie wystarczy do tego standardowy test znajdujący się w pamięci komputera. Najwięcej wad ujawnia się i najwięcej kłopotów występuje podczas transmisji danych. Konieczne jest więc wykonanie próby współpracy komputera z magnetofonem lub stacją. Polega to na odczytaniu dowolnej gry oraz zapisaniu, a następnie odczytaniu króciutkiego programu w Basicu. Najlepiej w tym celu skorzystać z pomocy kolegi lub znajomego, który taki sprzęt już posiada. Ponadto na giełdzie można niekiedy spotkać osoby dysponujące specjalnym modułem diagnostycznym, który pozwala na dość dokładne sprawdzenie komputera.

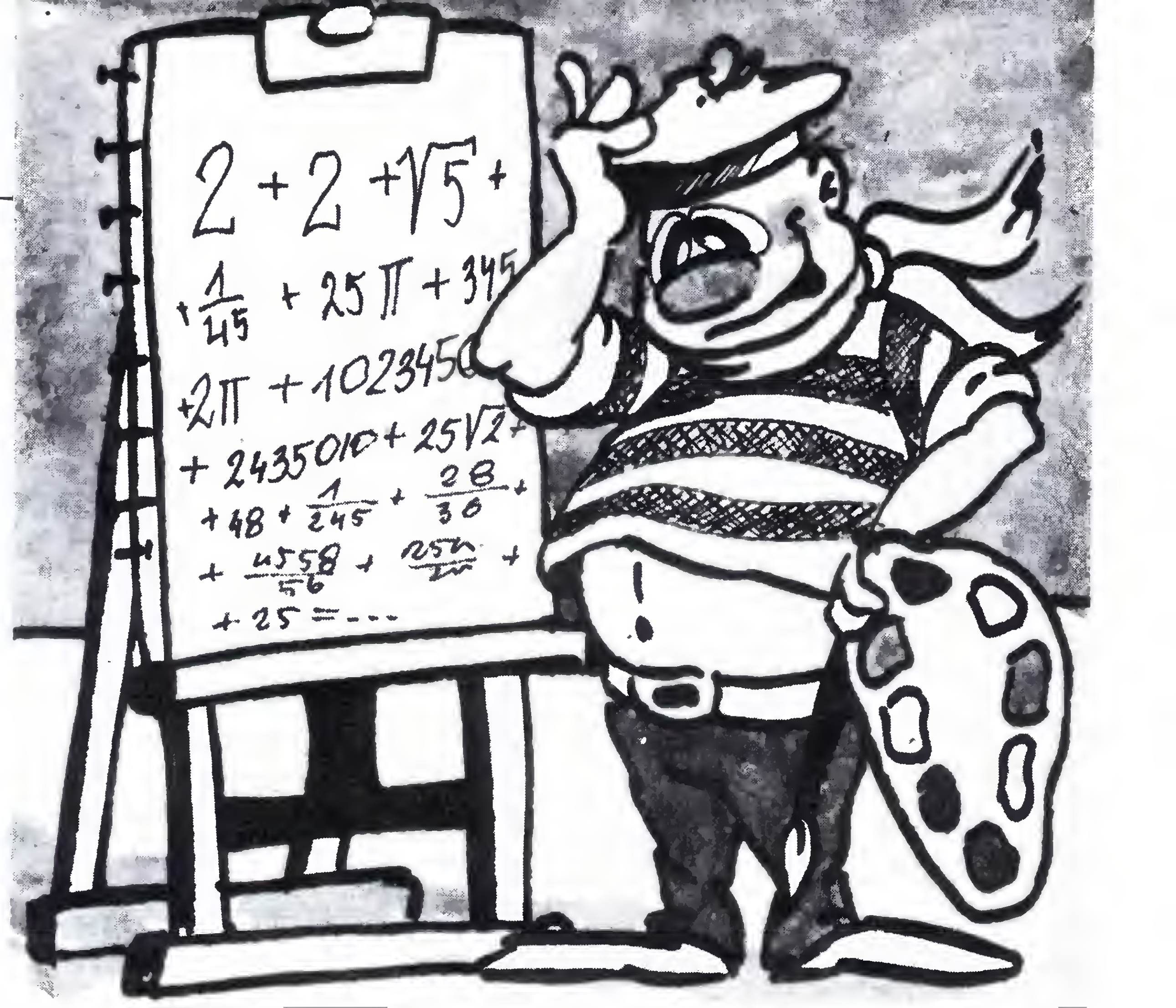
Niestety, żaden sposób nie zabezpieczy nas przed usterkami podobnymi do wcześniej opisanej. Jedynym sposobem uniknięcia kłopotu jest w takim przypadku posiadanie gwarancji. Przy zakupie na giełdzie należy się odpowiednio umówić ze sprzedawcą i zapisać jego personalia i adres. Nie da się tego zrobić w komisie, gdyż zwykle sklepy tego typu nie odpowiadają za wady ukryte towaru.

Życzę przyjemnych zakupów!

Wojciech Zientara



MATEMATYKA I SZTUKA



Wielu naszych Czytelników tworzy programy, które wykorzystują różne funkcje matematyczne i geometryczne do generowania obrazów. Nie jest to tylko zabawa, jak można by sądzić na pierwszy rzut oka. W rzeczywistości grafika „matematyczna" jest już dobrze rozwiniętą dziedziną informatyki. Po prostu, znacznie łatwiej zauważyć pewne rzeczy, gdy się je ogląda w postaci rysunku zamiast długich szeregów liczb. Takie właśnie programy matematyczno-graficzne będziemy od czasu do czasu publikować. Mamy przy tym nadzieję, że dzięki nim niektórzy inni Czytelnicy chociaż trochę polubią matematykę.

Wojciech Zientara

Poniższy program napisany w Atari Basic tworzy na ekranie efektowny ornament. Po uruchomieniu programu komputer zaczyna rysować linię generowaną losowo. Nie jest ona jednak chaotyczna. Zastosowana tu symetria osiowa i środkowa odbija się jakby w trzech lustrach, w wyniku czego powstaje niepowtarzalny wzór. Jedyny mankament programu to dosyć długi czas rysowania. Można więc zastosować Turbo Basic, co kilkakrotnie przyśpiesza rysowanie.

Klawisz stacji umożliwia zatrzymanie rysowania obrazu. Ponowne jego naciśnięcie powoduje kontynuację rysowania. Komputer będzie tworzył obraz praktycznie bez końca. Aby spowodować rysowanie obrazu od nowa należy przerwać program naciskając ‹BREAK› i ponownie go uruchomić. Oczywiście program można dowolnie zmieniać, aby uzyskać inny wygląd obrazu. Szczególnie duży wpływ na rysowany wzór mają zmiany parametrów instrukcji w wierszach 500 i 510.

Program może się wydawać nieco prymitywny, jednak dzięki niemu polubiłem przekształcenia na płaszczyźnie, które miałem na lekcjach matematyki.

Leszek Taratuta

```
ED    1 REM MATEMATYKA I SZTUKA
DF    2 REM Leszek Taratuta
IA    3 REM (c) 1990, Bajtek
NJ    4 REM
GX    10 GRAPHICS 8+16:COLOR 1:POKE 710,0
PS    20 GOSUB 500
OR    30 X=X+U
PU    40 GOSUB 500
PD    50 Y=Y+U
KQ    60 TRAP 200
OA    70 REM RYSOWANIE LINII
ZH    80 PLOT 160+X,95-Y
UQ    90 REM ODBICIE WZGLEDEM OSI Y
QF    100 PLOT 160-X,95-Y
QO    110 REM ODBICIE WZGLEDEM OSI X
NZ    120 PLOT 160+X,95+Y
UJ    130 REM SYMETRIA SRODKOWA
PD    140 PLOT 160-X,95+Y
LQ    150 IF PEEK(764)=33 THEN GOSUB 600
PY    160 GOTO 20
LM    200 IF X>159 THEN X=159
AL    210 IF X<-159 THEN X=-159
ZJ    220 IF Y>95 THEN Y=95
GZ    230 IF Y<-95 THEN Y=-95
PV    240 GOTO 20
QQ    500 U=INT(RND(0)*3)
II    510 U=U-1:RETURN
WM    600 POKE 764,255
IN    610 IF PEEK(764)<>33 THEN 610
XK    620 POKE 764,255:RETURN
```

KONFIGURACJA

Prezentuję dwie ciekawe procedury napisane w Action! służące do ustalania konfiguracji systemu (stacji dysków i edytora).

Procedura **Config** przeznaczona jest do zmiany konfiguracji stacji dysków. Kolejnymi jej parametrami są: liczba ścieżek na dyskietce, odstęp (przeplot) między sektorami, liczba sektorów na ścieżce, liczba stron dyskietki oraz liczba bajtów w sektorze. Dla standardowych gęstości należy stosować następujące parametry:

pojedyncza — **Config(40,20,18,1,128)**
rozszerzona — **Config(40,10,26,1,128)**
podwójna — **Config(40,20,18,1,256)**

Można także tworzyć własne niestandardowe formaty dyskietki (oczywiście pod warunkiem, że stacja potrafi je zrealizować). Na przykład format stosowany w komputerach IBM wywołuje się przez:

Config(40,40,9,1,512)

Takie nietypowe formaty można wykorzystać w celu zabezpieczenia dyskietki przed skopiowaniem.

Druga procedura — **Edytor** — steruje pracą edytora ekranowego. Kolejne jej parametry określają lewy i prawy margines obrazu oraz czas, jaki upływa od naciśnięcia klawisza do jego pierwszego powtórzenia i okres powtórzeń wciśniętego klawisza. Normalny stan edytora uzyskuje się po wywołaniu: **Edytor(2,32,40,5)**

Łukasz Komsta

```
PROC Config(BYTE sciezki,krok,sektory,
            strony CARD bajty)

 BYTE hi,ramtop=106
 TYPE DCB=[BYTE dev,num,cmd,stat,
          badr1,badr2,tout,unus
          CARD cntr,aux1]
 TYPE CON=[BYTE track,skew,sec1,sec2,
       page,den CARD bytes BYTE aux]
 DCB POINTER dptr=$300
 CON POINTER cn

  hi=ramtop-20
  cn=256*hi-1
  dptr.badr2=hi
  dptr.badr1=0
  cn.track=sciezki
  cn.skew=krok
  cn.sec2=sektory
  cn.page=strony
  cn.bytes=bajty
  cn.sec1=0
  cn.aux=$FF
  [32 89 228]
RETURN

PROC Edytor(BYTE lmarg,rmarg,keyd,keyr)

 TYPE MARGIN=[BYTE l,p]
 TYPE KEYV=[BYTE d,r]
 MARGIN POINTER marg=82
 KEYV POINTER key=729

  marg.l=lmarg
  marg.p=rmarg
  key.d=keyd
  key.r=keyr
RETURN
```

CP/M PLUS

— procedury systemowe w Turbo Pascalu (część druga)

W tej części prezentujemy 11 procedur pascalowych (listing 1) korzystających głównie z tablicy SCB (System Control Block) i trzech mniej znanych odwołań do modułu BDOS.

Funkcja GetSCBaddr, zwracająca adres tablicy SCB, była opisana w „Bajtku" 5/89. Trzy dalsze procedury (AssignCON, AssignAUX, AssignLST), także przedstawione we wspomnianym artykule, umożliwiają wzajemną zmianę przypisania niektórych urządzeń fizycznych i logicznych w systemie CP/M Plus.

Dwie procedury (Get i SetConsoleSize) pozwalają na pobranie i zadanie rozmiarów aktualnej konsoli, istotnej dla niektórych programów systemowych, np. DIR i TYPE.

Cztery nie używane bajty tablicy SCB mogą znaleźć dość wyrafinowane zastosowanie we własnych programach. Do tego celu służą dwie proste procedury Get4B i Set4B.

Ostatnia grupa funkcji i procedur ma zastosowanie przy współpracy z innymi programami. W szczególności kończący się program może przekazać kolejno uruchamianemu dwa bajty informacji dzięki parze Get i SetProgRetCode. Z kolei procedura ChainToProg pozwala z wnętrza własnego programu uruchomić inny program.

Demonstracją zestawu procedur jest krótki program przedstawiony na listingu 2.

Jonasz Mayer

LISTING 1 Plik SYS2.SYS

```
(*************************************************************)
(*                                                           *)
(*       Plik SYS2,SYS    (C) JM 1989                        *)
(*                                                           *)
(*  Biblioteka procedur systemowych w jezyku Turbo Pascal    *)
(*  przeznaczonych na komputery Amstrad PCW 8256/8512 i      *)
(*  CPC 6128, Czesc druga,                                   *)
(*                                                           *)
(* Lista procedur i funkcji:                                 *)
(*                                                           *)
(*   , Function  GetSCBaddr : integer;                       *)
(*                                                           *)
(*   , Procedure AssignCON (a : byte);                       *)
(*   , Procedure AssignAUX (a : byte);                       *)
(*   , Procedure AssignLST (a : byte);                       *)
(*                                                           *)
(*   , Procedure GetConsoleSize (Var Coulmn, Row : byte);    *)
(*   , Procedure SetConsoleSize (Column, Row : byte);        *)
(*                                                           *)
(*   , Procedure Get4B (Var b1,b2,b3,b4 : byte);             *)
(*   , Procedure Set4B (b1,b2,b3,b4 : byte);                 *)
(*                                                           *)
(*   , Function  GetProgRetCode : integer;                   *)
(*   , Procedure SetProgRetCode (code : integer);            *)
(*   , Procedure ChainToProg (command : _str80);             *)
(*,                                                          *)
(*************************************************************)
type
    _Str80 = String[80];

function SCBaddr : integer;
(**********************************)
(* Zwraca adres tablicy SCB,      *)
(**********************************)
var b : integer;
begin
     b := $003A;
     SCBaddr := BdosHL (49, addr(b));
end; (* of SCBaddr *)

const  {  Lista urzadzen fizycznych  }

  Cen = $10;  SIO = $20;  LPT = $40;  CRT = $80;

procedure AssignCON (a : byte);
(************************************************)
(* Przypisanie urzadzenia CONOUT do jednego z moz- *)
(* liwych urzadzen fizycznych (CEN/LPT/CRT/SIO,   *)
(************************************************)
begin
     Mem [SCBaddr+$25] := a;
end;  (* of Assign CON *)

procedure AssignAUX (a : byte);
(************************************************)
(* Przypisanie urzadzenia AUXOUT do jednego z moz- *)
(* liwych urzadzen fizycznych (CEN/LPT/CRT/SIO,   *)
(************************************************)
begin
     Mem [SCBaddr+$29] := a;
end;  (* of Assign AUX *)

procedure AssignLST (a : byte);
(************************************************)
(* Przypisanie urzadzenia LSTOUT do jednego z moz- *)
(* liwych urzadzen fizycznych (CEN/LPT/CRT/SIO,   *)
(************************************************)
begin
     Mem [SCBaddr+$2B] := a;
end;  (* of Assign Lst *)

Procedure GetConsoleSize (var Column, Row : byte);
(************************************************)
(* Pobranie aktualnych rozmiarow konsoli,       *)
(************************************************)
begin
   Column := Mem [SCBaddr+$1A];
   Row    := Mem [SCBaddr+$1C];
end; (* of Get Console Size *)

Procedure SetConsoleSize (Column, Row : byte);
(************************************************)
(* Zadanie aktualnych rozmiarow konsoli,        *)
(************************************************)
begin
   Mem [SCBaddr+$1A] := Column;
   Mem [SCBaddr+$1C] := Row;
end; (* of Set Console Size *)

Procedure Get4B (var b1,b2,b3,b4 : byte);
(************************************************)
(* Pobranie z tablicy SCB 4 bajtow niezywanych  *)
(* przez system,                                *)
(************************************************)
var s : integer;
begin
   s := SCBaddr;
   b1 := Mem [s+$06];
   b2 := Mem [s+$07];
   b3 := Mem [s+$08];
   b4 := Mem [s+$09];
end; (* of Get 4 bytes *)

Procedure Set4B (b1,b2,b3,b4 : byte);
(************************************************)
(* Zadanie 4 bajtow w tablicy SCB,              *)
(************************************************)
var s : integer;
begin
   s := SCBaddr;
   Mem [s+$06] := b1;
   Mem [s+$07] := b2;
   Mem [s+$08] := b3;
   Mem [s+$09] := b4;
end; (* of Set 4 bytes *)

Function  GetProgRetCode : integer;
(************************************************)
(* Pobranie kodu powrotnego poprzedniego programu, *)
(************************************************)
begin
   GetProgRetCode := BDOSHL ($6C,$FFFF);
end; (* of Get Program Return Code *)

Procedure SetProgRetCode (Code : integer);
(************************************************)
(* Zadanie kodu powrotnego dla nastepnego programu,*)
(************************************************)
begin
   BDOS ($6C,Code);
end; (* of Set Program Return Code *)

Procedure ChainToProg (command : _str80);
(************************************************)
(* Wykonanie polecenia systemowego po zakonczeniu *)
(* egzekucji programu,                          *)
(************************************************)
var L : byte;
begin
   L := Length (Command);
   Move (Command, Mem [$80],L);
   Mem [$80+L] := 0;
   BDOS ($2F,$00);
end; (* of Chain To Program *)

(************************************************
```

LISTING 2 Plik DEMOSYS2.PAS

```
Program SystemDemo2;
(****************************************************************)
(*                                                              *)
(*        Plik DEMOSYS2,PAS     (C) JM 1989                     *)
(*                                                              *)
(*   Demonstruje dzialanie biblioteki systemowej SYS2,SYS,      *)
(*                                                              *)
(****************************************************************)

{$I SYS2,SYS }

var
   flag, x,y, dummy : byte;
   ch      : char;

begin
     ClrScr;
     Get4B (flag,x,y,dummy);
     if flag=0
     then begin
            writeln ('Pierwszy przebieg programu DEMOSYS2');
            writeln;
            writeln ('Rozmiar konsoli, istotny dla niektorych programow ');
            writeln ('systemowych (DIR, TYPE) zostanie zmiejszony,');
            writeln ('Po zakonczeniu dzialania programu zostanie uruchomiony ');
            writeln ('program DIR,COM, ');
            writeln ('Nastepnie nalezy ponownie wykonac program DEMOSYS2,');
            writeln;
            GetConsoleSize (x,y);
            Set4B (1,x,y,0);
            SetConsoleSize (10,10);
            writeln('Nacisnij klawisz,');
            repeat until keypressed; read (kbd,ch);
            ClrScr;
            ChainToProg ('dir a;[a'#13);
          end
     else begin
            writeln ('Drugi przebieg programu DEMOSYS2');
            writeln ('Przywrocenie pierwotnego rozmiaru konsoli,');
            Set4B (0,0,0,0);
            SetConsoleSize (x,y);
            writeln('Nacisnij klawisz,');
            repeat until keypressed; read (kbd,ch);
            ClrScr;
            ChainToProg ('dir a;[a'#13);
          end;
end.
(****************************************************************)
```

RSX dla każdego

część druga i ostatnia

W pierwszej części został opisany sposób podłączenia RSX-a oraz metody przekazywania parametrów z BASIC-a do RSX-a. Część druga ujawni następne informacje: jak przekazać wyniki RSX-a do programu w BASIC-u i jak wywołać RSX-a z programu w kodzie maszynowym.

A w drugą stronę?

A w drugą stronę? Jak przekazać coś z RSX-a do BASIC-a? Jest to oczywiście możliwe, choć trochę trudniejsze niż przekazanie w odwrotną stronę. Nie można przekazać wyniku w sposób bezpośredni, trzeba go skopiować do zmiennej. Ba, ale jak znaleźć zmienną gdzieś w pamięci? Nie jest aż tak źle, nie trzeba szukać. Adres zmiennej musi zostać przekazany jako parametr. W tym przypadku korzystamy z funkcji „@" BASIC-a: wyrażenie: „@nazwa" daje adres zmiennej „nazwa". Adres zmiennej uzyskany w ten sposób wskazuje na pole danych, czyli: dwa bajty wartości (zmienne całkowite), pięć bajtów wartości (zmienne rzeczywiste) lub deskryptor łańcucha. Możemy także sprawdzić, czy przekazana do procedury zmienna jest właściwego typu (czego nie można sprawdzić w przypadku innych parametrów). Badając bajt o adresie o 1 niższym niż dany, możemy otrzymać wartości: 1 — zmienna całkowita, 2 — łańcuch lub 4 — zmienna rzeczywista. Przy umieszczaniu wyniku w zmiennej trzeba uważać, by nie przekroczyć rozmiaru pola danych, co spowodowałoby zniszczenie tablicy zmiennych. W przypadku łańcuchów dane kopiujemy w miejsce poprzedniej zawartości (wskazywanej) przez adres w deskryptorze), uważając, by nie przekroczyć długości (także podana w deskryptorze). Oczywiście (chyba niepotrzebnie to piszę) zmienna, którą przekazujemy, do RSX-a, musi istnieć (być co najmniej raz użyta).

Tylko dla asembleromaniaków!

Czy można wywołać RSX z programu w kodzie maszynowym? Czemu nie! Nie jest to takie proste, ale wykonalne. Po pierwsze, trzeba znaleźć adres procedury RSX-a. Jeśli w parze HL umieścimy adres nazwy RSX-a i wywołamy procedurę KL FIND COMMAND (#BCD4), to otrzymamy HL zawierające adres procedury RSX-a i C zawierające kod ROM (nr ROM-u zawierającego prodecurę RSX-a) oraz ustawioną flagę CARRY. Jeśli nie znaleziono rozkazu RSX o podanej nazwie, to CARRY będzie skasowane. Po drugie, trzeba przygotować parametry, umieścić je gdzieś (w odwrotnej kolejności!), ustawić IX na adres ostatniego parametru (para DE powinna zawierać wartość tego parametru) i umieścić w akumulatorze liczbę parametrów. Po trzecie, należy wykonać rozkaz „CALL #1B" (mając HL i C ustawione w pierwszym kroku). Pod adresem #1B znajduje się procedura FAR PCHL, która skacze pod adres podany w rejestrach HL i C.

A przykłady?

Bez obawy, są przykłady. Listingi 1 i 2 to rozwinięcie procedur z artykułu „Dodatkowa pamięć RAM" (tylko dla CPC 6128). Listing 1 to procedura w asemblerze, a nr 2 to jej wersja po asemblacji i przerobieniu na BASIC. Otrzymujemy trzy rozkazy RSX: „|ZAPIS", „|LADUJ" i „|POKAZ". Rozkaz „|ZAPIS, nr. bloku" zapamiętuje obrazek w drugim banku, a „|LADUJ, nr. bloku" i „|POKAZ, nr. bloku" ponownie umieszczają zapamiętany obrazek na ekranie. „|POKAZ" robi to w nietypowy sposób, jest to zwykły „bajer" (ale jaki fajny!). Polega to na powtarzaniu w pętli kopiowania bajtów, ale nie kolejno (jak to robi „|LADUJ"), ale z losowym krokiem, ustalanym na początku.

Listingi 3 i 4 prezentują przekazywanie danych do BASIC-a. Rozkaz „|GET.INK, nr @x%" umieszcza w zmiennej „x%" kolor pisaka o podanym numerze — jest to odwrotność rozkazu „INK". Podobnie „|GET.BORDER, @x%" odczytuje kolor ramki. Po wykonaniu jednego z tych rozkazów zmienna „x%" otrzyma wartość „pierwszy kolor + 256 * drugi kolor". Do kompletu jest rozkaz „|GET.MODE, @x%", który odczytuje aktualny tryb ekranowy.

Listing 5 to przykład wywoływania RSX-a z kodu maszynowego: prosta procedura wybierania stacji dysków, wykorzystująca rozkaz |DRIVE. Na wejściu akumulator zawiera nazwę stacji („A" lub „B").

Dodatek: Idziemy z postępem!

Listing 6 zawiera tekst programu, który przerabia stare programy, z funkcjami wywoływanymi przez „CALL", na zestaw RSX-ów. Podajemy nazwy rozkazów i adresy procedur, a program dopisuje nagłówek definiujący RSX-y. A więc, zamiast mrowia adresów do zapamiętania, przyjemne i zrozumiałe rozkazy. Skrócona instrukcja obsługi: załaduj procedury w kodzie maszynowym, uruchom program „Make RSX" i sensownie odpowiadaj na pytania.

Odtąd obowiązuje zasada: jeśli są więcej niż trzy procedury w kodzie maszynowym, muszą być one w postaci RSX-ów. Łamiących to prawo kara nie ominie.

Michał Szokoło

LISTING 1

```
10  #LIST     ON
    #COUNT    ON
   ;
   ; "RSX dla kazdego" cz.2 - Listing 3
   ; Copyright (c)1990 by SEM 03375991 B
   ;
          ORG   #A400
   ;
20 instal:
          LD    BC,tablica
          LD    HL,kolejka
          CALL  KL_LOG_EXT
   inster:
          LD    C,#FD ; generacja bledu
          CALL  HIKL_PROBE_ROM
          LD    A,L
          CP    0 ; czy CPC 464?
          RET   NZ
          LD    HL,BAS_ERR_464 ; adres proc.
          LD    (bladl),HL ; bledu
          RET
30 tablica:
          DW    nazwy
          JP    get_ink
          JP    get_border
          JP    get_mode
   nazwy: DB    "GET.IN","K"+128
          DB    "GET.BORDE","R"+128
          DB    "GET.MOD","E"+128
          DB    0
   kolejka:
          DS    4
   adr_zm:
          DW    0
   bladl: DW    BAS_ERR
          DB    #FD
40 sprawdz:
   ; czy podano adres zmiennej calk.?
          LD    (adr_zm),DE ; zapamietaj adres
          DEC   DE
          LD    A,(DE) ; bajt typu
          CP    1 ; integer?
          RET   Z ; tak, wykonuj dalej
   blad : LD    A,33 ; blad BASIC-a
          LD    E,A
          RST   #18 ; wywolaj proc. BASIC-a
          DW    bladl
          RET
50 get_border:
          CP    1 ;jeden parametr?
          JP    NZ,blad
          CALL  sprawdz
          CALL  SCR_GET_BORDER
   putcol:
          LD    HL,(adr_zm)
          LD    (HL),B
          INC   HL
          LD    (HL),C
          RET
60 get_mode:
          CP    1
          JP    NZ,blad
          CALL  sprawdz
          CALL  SCR_GET_MODE
          LD    HL,(adr_zm)
          LD    (HL),A
          INC   HL
          LD    (HL),0
          RET
70 get_ink:
          CP    2
          JP    NZ,blad
          CALL  sprawdz
          LD    A,(IX+2)
          AND   #F
          CALL  SCR_GET_INK
          JR    putcol
80 ; adresy procedur systemowych
   KL_LOG_EXT:
          EQU   #BCD1
   SCR_GET_MODE:
          EQU   #BC11
   SCR_GET_INK:
          EQU   #BC35
   SCR_GET_BORDER:
          EQU   #BC3B
   HIKL_PROBE_ROM:
          EQU   #B915
   BAS_ERR:
          EQU   #CB55
   BAS_ERR_464:
          EQU   #CA94
```

LISTING 2

```
1000 '
1010 ' RSX dla kazdego Cz.2 - Listing 2
1020 ' (c)1990 by SEM 03375991 B
1030 '
1040 MODE 1 : CALL &BC02 : PEN 1 : PAPER 0
1050 PRINT "RSX DLA KAZDEGO Cz.II - Listing 2"
1060 PRINT
1070 GOTO 1180
1080 DATA 01,13,a5,21,0f,a5,cd,d1,bc,3e,c9,32,00,a5,c9,00
1090 DATA 00,00,00,1e,a5,c3,2e,a5,c3,41,a5,c3,54,a5,5a,41
1100 DATA 50,49,d3,4c,41,44,55,ca,50,4f,4b,41,da,00,fe,01
1110 DATA c0,cd,7c,a5,21,00,c0,11,00,40,01,00,40,ed,b0,18
1120 DATA 37,fe,01,c0,cd,7c,a5,21,00,40,11,00,c0,01,00,40
1130 DATA ed,b0,18,24,fe,01,c0,cd,7c,a5,ed,5f,cb,c7,16,00
1140 DATA 5f,21,00,40,7e,cb,fc,77,19,cb,bc,cb,f4,7d,b7,20
1150 DATA f3,7c,fe,40,20,ee,18,00,af,c3,5b,bd,7b,e6,03,f6
1160 DATA c4,c3,5b,bd
1170 DATA 14964
1180 SYMBOL AFTER 256 : MEMORY &A4FF
1190 RESTORE 1080 : sum=0
1200 FOR adr=&A500 TO &A583
1210     READ a$
1220     b=VAL("&"+a$)
1230     sum=(sum+b)MOD &4000
1240     POKE adr,b
1250 NEXT adr
1260 READ b
1270 IF b<>sum THEN PRINT"DATA ERROR!" : STOP
1280 CALL &A500 : POKE &A500,&C9
1290 PRINT "  Nowe rozkazy:"
1300 PRINT "     |ZAPIS,nr.banku"
1310 PRINT "     |LADUJ,nr.banku"
1320 PRINT "     |POKAZ,nr.banku"
1330 PRINT
1340 NEW
1350 REM koniec, the end, fin, ende...
```

LISTING 3

```
10  *LIST    ON
    *COUNT   ON
    ;
    ; "RSX dla kazdego" cz.2 - Listing 1
    ; Copyright (c)1990 by SEM 03375991 B
    ;
            ORG  #A500
    ;
20 instal:
            LD   BC,tablica
            LD   HL,wolne
            CALL KL_LOG_EXT
            LD   A,#C9 ; kod "RET"
            LD   (instal),A ; zablokuj instalator
            RET
30 wolne:   DS   4
    tablica:
            DW   nazwy
            JP   zapam
            JP   laduj
            JP   pokaz
    nazwy:  DB   "ZAPI","S"+128
            DB   "LADU","J"+128
            DB   "POKA","Z"+128
            DB   0
40 zapam:   ; obrazek do pamieci
            CP   1
            RET  NZ
            CALL przygotuj
            LD   HL,#C000 ;z ekranu
            LD   DE,#4000 ; do pamieci
            LD   BC,#4000 ; 16 K
            LDIR  ; przeslij
            JR   standard
    laduj:  ; obrazek na ekran
            CP   1
            RET  NZ
            CALL przygotuj
            LD   HL,#4000 ; z pamieci
            LD   DE,#C000 ; na ekran
            LD   BC,#4000 ; 16 K
            LDIR  ; przeslij
            JR   standard
50 pokaz:   CP   1
            RET  NZ
            CALL przygotuj
            LD   A,R ; wartosc losowa
            SET  0,A ; ustaw nieparzyste
            LD   D,0
            LD   E,A
            LD   HL,#4000
    petla:  LD   A,(HL) ; wez bajt
            SET  7,H ; adres #Cxxx
            LD   (HL),A ; na ekran
            ADD  HL,DE ; nowy adres
            RES  7,H ; adres #4xxx
            SET  6,H
            LD   A,L
            OR   A ; nizszy bajt=0?
            JR   NZ,petla ; jesli nie, to dalej
            LD   A,H
            CP   #40 ; wyzszy bajt=#40?
            JR   NZ,petla
            JR   standard
60 standard:
    ; konfig. standard i powrot
            XOR  A
            JP   KL_RAM_CONFIG
    przygotuj:
    ; oblicz i przelacz konfiguracje
            LD   A,E ;ostatni parametr
            AND  3
            OR   #C4
            JP   KL_RAM_CONFIG
70 ; Adresy procedur systemowych
    KL_RAM_CONFIG:
            EQU  #BD5B
    KL_LOG_EXT:
            EQU  #BCD1
```

LISTING 4

```
1000 '
1010 ' RSX DLA KAZDEGO Cz.II - Listing 4
1020 ' (c)1990 by SEM 03375991 B
1030 '
1040 MODE 1 : CALL &BC02 : PEN 1 : PAPER 0
1050 PRINT "RSX DLA KAZDEGO Cz.II - Listing 4"
1060 PRINT
1070 GOTO 1180
1080 DATA 01,19,a4,21,3e,a4,cd,d1,bc,0e,fd,cd,15,b9,7d,fe
1090 DATA 00,c0,21,94,ca,22,44,a4,c9,24,a4,c3,7c,a4,c3,57
1100 DATA a4,c3,69,a4,47,45,54,2e,49,4e,cb,47,45,54,2e,42
1110 DATA 4f,52,44,45,d2,47,45,54,2e,4d,4f,44,c5,00,00,00
1120 DATA 00,00,00,00,55,cb,fd,ed,53,42,a4,1b,1a,fe,01,c8
1130 DATA 3e,21,5f,df,44,a4,c9,fe,01,c2,50,a4,cd,47,a4,cd
1140 DATA 3b,bc,2a,42,a4,70,23,71,c9,fe,01,c2,50,a4,cd,47
1150 DATA a4,cd,11,bc,2a,42,a4,77,23,36,00,c9,fe,02,c2,50
1160 DATA a4,cd,47,a4,dd,7e,02,e6,0f,cd,35,bc,18,d4,1c
1170 DATA 16327
1180 SYMBOL AFTER 256 : MEMORY &A3FF
1190 RESTORE 1080 : sum=0
1200 FOR ADR=&A400 TO &A48E
1210     READ a$
1220     b=VAL("&"+a$)
1230     sum=(sum+b)MOD &4000
1240     POKE adr,b
1250 NEXT adr
1260 READ b
1270 IF b<>sum THEN PRINT"DATA ERROR!":STOP
1280 CALL &A400 : POKE &A400,&C9
1290 PRINT "  Nowe rozkazy:"
1300 PRINT "     |GET.INK,nr,@x%"
1310 PRINT "     |GET.BORDER,@x%"
1320 PRINT "     |GET.MODE,@x%"
1330 PRINT
1340 NEW
1350 REM koniec, the end, fin, ende...
```

LISTING 5

```
10 ;
   ; RSX dla kazdego Cz.2 - Listing 5
   ; (c)1990 by SEM 03375991 B
   ;
20 ;--- Kod procedury ---
   stacja:
           LD   (nazwa_stacji),A
           LD   HL,drive_rsx
           CALL KL_FIND_COMMAND
           LD   IX,deskryptor
```

```
           LD   A,1
           CALL FAR_PCHL
           RET
30 stacja_a:
           LD   A,"A"
           JR   stacja
   stacja_b:
           LD   A,"B"
           JR   stacja
40 ;--- Procedury systemowe ---
   KL_FIND_COMMAND:
           EQU  #BCD4
   FAR_PCHL:
           EQU  #1B
   ;--- Zmienne procedury ---
   drive_rsx:
           DEFB "DRIV","E"+128
   nazwa_stacji:
           DEFB "A"
   deskryptor:
           DEFB 1
           DEFW nazwa
   ;--- Koniec ---
```

LISTING 6

```
10 '
20 ' Make RSX (c)1990 SEM 03375991 B
30 ' RSX dla kazdego Cz.2 - Listing 6
40 ' (c)1990 by SEM 03375991 B
50 '
60 DEF FN adres(adr)=-adr*(adr>=0)-(65536+adr)*(adr<0)
70 DEF FN hi(adr)=INT(adr/256)
80 DEF FN lo(adr)=adr-256*FN hi(adr)
90 DEF FN cyfra(x$)=(x$>="0") AND (x$<="9")
100 MODE 2 : CALL &BC02 : PAPER 0 : PEN 1
110 PRINT CHR$(24);" Make RSX ";CHR$(24);" (c)1990 by SEM 03375991 B"
120 WINDOW 1,80,3,25
130 INPUT "Ile rozkazow";ilosc : IF ilosc<=0 THEN 130
140 PRINT
150 DIM nazwy$[ilosc],adresy[ilosc]
160 FOR nr=1 TO ilosc
170     GOTO 190
180     PRINT CHR$(7);CHR$(24);" Blad! Powtorz! ";CHR$(24)
190     PRINT USING "###: ";nr;
200     INPUT "Podaj rozkaz i adres procedury: ",r$,a
210     IF FN cyfra(LEFT$(r$,1)) OR LEN(r$)>16 THEN GOTO 180
220     nazwy$[nr]=UPPER$(r$):adresy[nr]=FN adres(a)
230 NEXT nr
240 dlugosc=1
250 FOR nr=1 TO ilosc
260     r$=nazwy$[nr]
270     r=ASC(RIGHT$(r$,1))
280     r=r OR 128
290     MID$(r$,LEN(r$),1)=CHR$(r)
300     dlugosc=dlugosc+LEN(r$)
310     nazwy$[nr]=r$
320 NEXT nr
330 dlnagl=dlugosc+15+ilosc*3
340 PRINT : PRINT "Dlugosc naglowka RSX:";dlnagl;"bajtow." : PRINT
350 INPUT "Podaj adres pierwszego bajtu twojego programu: ",start
360 INPUT "Podaj dlugosc twojego programu: ",dlprog
370 start=FN adres(start)
380 nowyst=start-dlnagl : IF HIMEM>=nowyst THEN MEMORY nowyst-1
390 mem=start-dlugosc : skoki=mem-(3*ilosc) : rozkazy=mem
400 FOR nr=1 TO ilosc
410     r$=nazwy$[nr]
420     FOR i=1 TO LEN(r$)
430         POKE mem,ASC(MID$(r$,i))
440         mem=mem+1
450     NEXT i
460 NEXT nr
470 POKE mem,0
480 FOR nr=0 TO ilosc-1
490     i=skoki+(3*nr)
500     POKE i,&C3
510     POKE i+1,FN lo(adresy[nr+1])
520     POKE i+2,FN hi(adresy[nr+1])
530 NEXT nr
540 tabsk=skoki-2
550 POKE tabsk+1,FN hi(rozkazy)
560 POKE tabsk,FN lo(rozkazy)
570 podl=skoki-6
580 DATA 1,sk,33,pd,&c3,&d1,&bc,*
590 mem=nowyst
600 READ a$
610 WHILE a$<>"*"
620    IF a$="sk" THEN POKE mem,FN lo(tabsk) : POKE mem+1,FN hi(ta  :)
630    IF a$="pd" THEN POKE mem,FN lo(podl) : POKE mem+1,FN hi(pod
640    IF VAL(a$)<>0 THEN POKE mem,VAL(a$) : mem=mem+1 ELSE mem=me  2
650    READ a$
660 WEND
670 PRINT
680 PRINT "Nowy adres ladowania:" ; nowyst ; "= &" ; HEX$(nowyst,4)
690 PRINT "Nowa dlugosc programu:" ; dlnagl+dlprog
700 PRINT
710 PRINT "Po zaladowaniu wykonaj rozkaz: CALL &" ; HEX$(nowyst,4)
720 PRINT
730 INPUT "Pod jaka nazwa nagrac";n$
740 SAVE "!"+n$,b,nowyst,dlnagl+dlprog
750 END
```

PROCEDURY SYSTEMOWE AMSTRADA
(część IV)

PAKIET KODU MASZYNOWEGO (MACHINE CODE) — dokończenie

MC CLEAR INKS — USTAW JEDNOLITY KOLOR #BD22

Ustawia wszystkie kolory na jednakową wartość.
Wejście: w DE adres bajtu zawierającego kod koloru (sprzętowy).
Wyjście: AF skasowane.

MC SET INKS — USTAW KOLORY #BD25

Ustawia wszystkie kolory według tablicy.
Wejście: DE zawiera adres tablicy kolorów.
Wyjście: AF skasowane.
Format tablicy kolorów: bajt 0 określa kolor ramki (borderu), bajty 1-16 określają kolory pisaków (PEN) 0-15.
Uwaga: tablica zawiera sprzętowe kody kolorów.

MC RESET PRINTER — INICJALIZUJ OBSŁUGĘ DRUKARKI #BD28

Odtwarza skoki pośrednie (indirections) procedur obsługi drukarki.
Wejście: nie ma.
Wyjście: AF, HL, DE i BC skasowane.

MC PRINT CHAR — DRUKUJ ZNAK NA DRUKARCE #BD2B

Wysłanie znaku na drukarkę. Jeśli drukarka nie jest gotowa, procedura czeka przez około 0.4 sek. na zgłoszenie gotowości.
Wejście: akumulator zawiera znak (bit 7 jest pomijany).
Wyjście: AF zawsze skasowane. Jeśli znak został wysłany, to CY=1, jeśli wysłanie było niemożliwe, to CY=0.

MC BUSY PRINTER — SPRAWDŹ GOTOWOŚĆ DRUKARKI #BD2E

Sprawdza, czy drukarka jest gotowa do przyjęcia znaku (sygnał BUSY nieaktywny).
Wejście: nie ma.
Wyjście: flagi skasowane. Jeśli drukarka jest gotowa, to CY=0, jeśli nie jest gotowa, to CY=1.

MC SEND CHAR — WYŚLIJ ZNAK NA DRUKARKĘ #BD31

Wyślij znak na drukarkę, nie sprawdzając gotowości.
Wejście: akumulator zawiera znak do wydrukowania (bit 7 pomijany).
Wyjście: AF skasowane, CY=1 (operacja jest zawsze udana)

MC SOUND REGISTER — ZAŁADUJ REJESTR DŹWIĘKU #BD34

Umieszcza podany bajt w podanym rejestrze układu AY-3-8912 (PSG). Konieczna jest znajomość funkcji rejestrów PSG.
Wejście: akumulator zawiera numer rejestru, C zawiera daną.
Wyjście: AF i BC skasowane.

MC JUMP RESTORE — ODTWÓRZ SKOKI #BD37

Odtwarza tablicę skoków systemowych.
Wejście: nie ma.
Wyjście: AF, BC, DE i HL skasowane.

KONIEC GŁÓWNEJ TABLICY SKOKÓW

Dalsze pozycje w tablicy skoków zależne są od typu komputera, są to m.in. skoki pakietu edytora, dodatkowe procedury w CPC-6128 i CPC-664 oraz tzw. skoki pośrednie (indirections). W tym odcinku znajdą się niektóre dodatkowe procedury CPC-664/6128, nie wprowadzone w CPC-464.

PROCEDURY DODATKOWE

KM SET STATE — USTAW STAN CAPS LOCK I SHIFT LOCK #BD3A

Ustawia flagi CAPS LOCK i SHIFT LOCK. Ustawienie może zostać zmienione z klawiatury.
Wejście: H zawiera stan CAPS LOCK, L zawiera stan SHIFT LOCK. Zero oznacza wyłączenie, 255 — włączenie.
Wyjście: wszystkie rejestry zachowane.

KM EMPTY BUFFER — OPRÓŻNIJ BUFOR KLAWIATURY #BD3D

Opróżnia bufor klawiatury (typeahead buffer).
Wejście: nie ma.
Wyjście: AF skasowane.

TXT GET CURSOR FLAG — ODCZYTAJ STAN KURSORA #BD40

Odczytuje stan kursora (czy jest widoczny).
Wejście: nie ma.
Wyjście: akumulator zawiera bajt stanu:
- bit 0: flaga użytkownika
- bit 1: flaga systemowa

Wartość „1" oznacza zakaz. Kursor jest widoczny gdy A=0.

C.D.N.

Ten odcinek jest nieco krótszy od poprzednich, ponieważ niespodziewanie natrafiłem na „dziurę" w dokumentacji. Nie martwcie się, już niedługo następny odcinek, oparty na „SOFT 968" (firmowy opis systemu). Nie będzie żadnych luk, dziur itp.

Michał Szokoło

TIMEX i AUTO-FIRE

Jedną z fabrycznych wad TIMEX'a jest błędna interpretacja sygnału AUTO-FIRE wysyłanego przez niektóre typy joysticków.

Komputer ten został wyposażony w gniazdo dla joysticka, układ znajdujący się wewnątrz komputera działa podobnie jak interface systemu Kempston podłączanego do ZX Spectrum. Dlaczego jednak działanie ich nie jest identyczne? Otóż dzieje się tak dlatego, że w TIMEX'ie masa joysticka jest podpięta przez opornik 1.5 kΩ (R21) do **+5V**, a linie sygnałowe (góra, dół, lewo, prawo, strzał oraz trzy nie wykorzystane) zwarte do masy za pośrednictwem oporników 10 kΩ (R5÷12). W związku z tym zwarcie styku powoduje podanie na wejście bufora **74-LS-244** logicznej „1" i wszystko jest w porządku, aż do momentu, gdy włączymy AUTO-FIRE. W joysticku przy różnicy potencjału **U.. — GND** rzędu 0.5V żaden generator impulsu nie zadziała, a efekt jest taki, jakby wciśnięto i przytrzymano przycisk strzału — FIRE.

Może jest to użyteczne dla np. ART STUDIO, lecz nie dla gier takich jak R-TYPE.

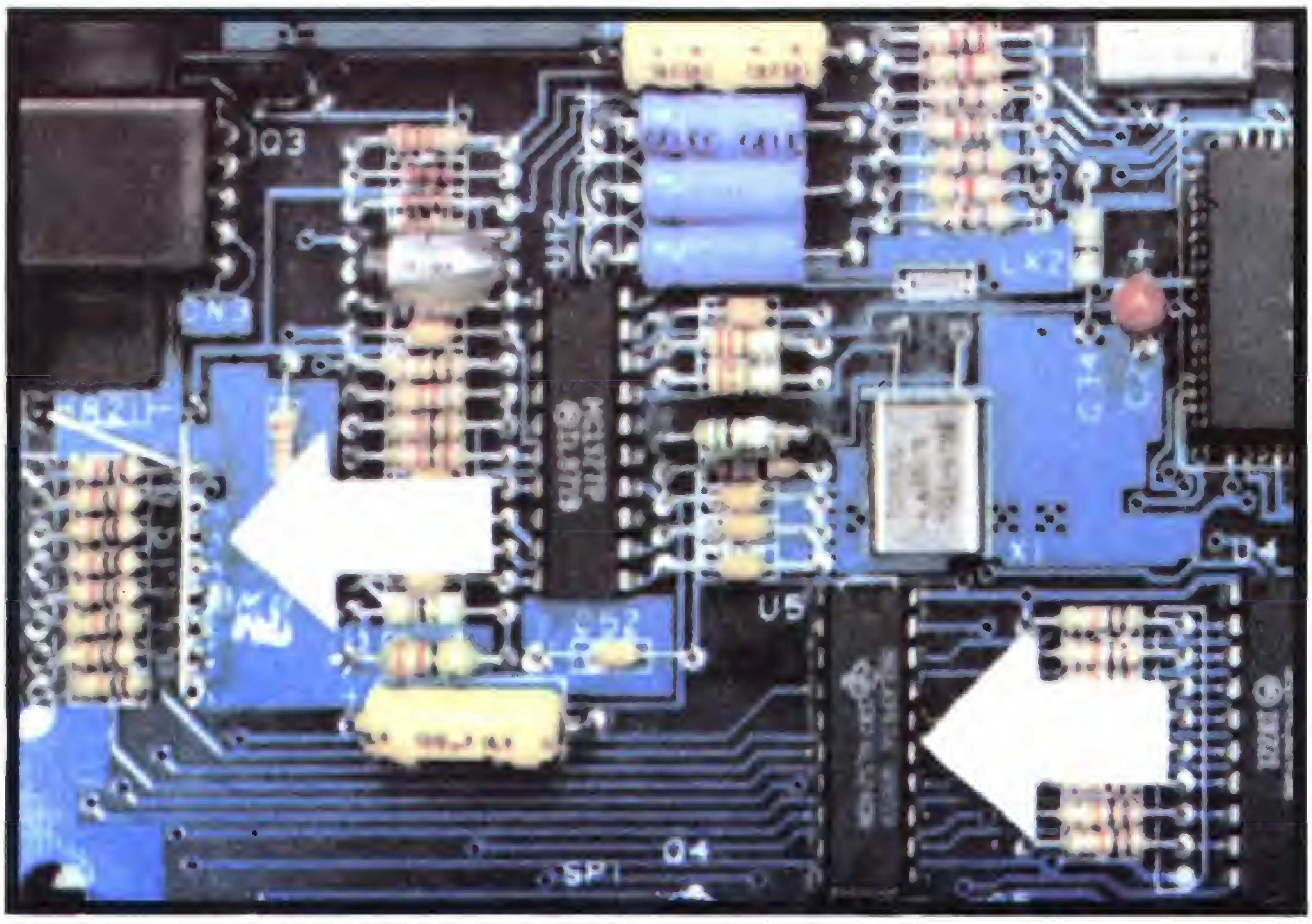
Celem mojej przeróbki jest „odblokowanie" generatora impulsów przy zachowaniu normalnego odczytu stanu linii sygnałowych Kempstona.

W przyjętym przeze mnie rozwiązaniu opornik **R21** został usunięty, a otwory lutownicze po nim zostały wykorzystane do podłączenia **0V** na masę joysticka. Otwory te używam również jako źródło **+5V** do „podciągnięcia" napięcia linii sygnałowych poprzez wylutowane prawym końcem z masy oporniki **R5÷12** i zwarte z tymże napięciem. Aby zwarciu styku towarzyszyła „1" na odpowiedniej szynie danych, konieczne jest zastąpienie układu **74-LS-244** jego odpowiednikiem z wyjściami trójstanowymi zanegowanymi: **74-LS-240**.

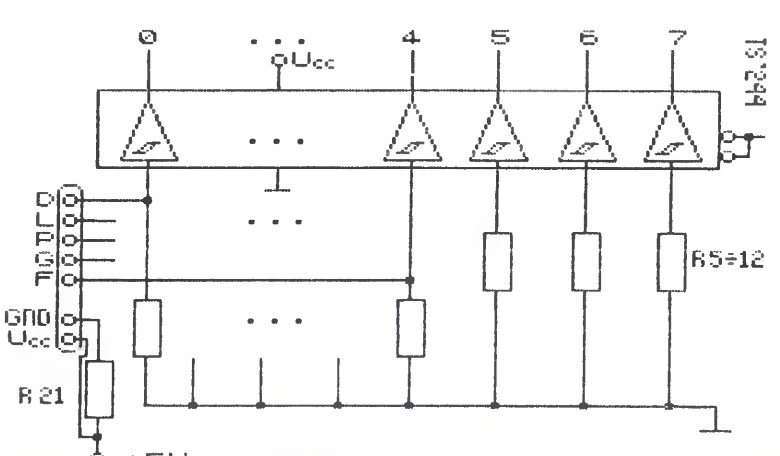
Przeróbka jest prosta, aczkolwiek wymaga dużej staranności i wprawy w posługiwaniu się lutownicą.

Zamiennikami układu **74-LS-240** mogą być: **'340**, **'540** oraz **'640** wykonane niekoniecznie w technologii TTL-LS (układ pracuje jako nadajnik, nie obciąża procesora).

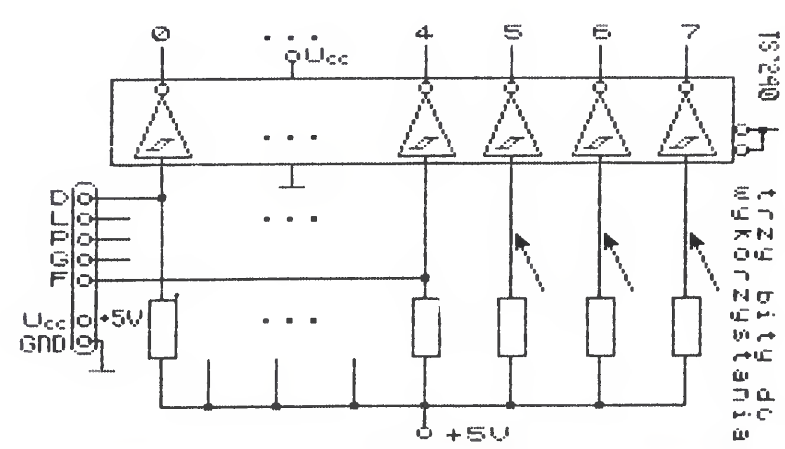
Mieczysław Zalech
(student I roku elektroniki)



Widok przerobionego układu



Układ przed przeróbką



Układ po przeróbce

ROZSZERZENIE FUNKCJI SCREEN$

Prezentowane w artykule zagadnienie jest przykładem sztuczki przydatnej podczas pisania samodzielnych programów w BASIC-u.

Czasem zdarza się, że rozwiązanie trudnych z pozoru problemów bywa bardzo proste. Tak też się stało w przypadku listu czytelnika, który chciał, aby funkcja **SCREEN$** „nauczyła" się rozpoznawać znaki grafiki **UDG**. Dodatkowo należało ją do tego przekonać posługując się wyłącznie BASIC-em. Takie ograniczenia spowodowały, że początkowo było to zadanie niewykonalne.

Jednak analiza funkcji **SCREEN$** podsunęła najprostsze i chyba jedyne rozwiązanie: wystarczą dwa **POKE'i**, aby ominąć jej ograniczenia. Pokazuje to poniższy program.

```
 10 REM ********************
 20 REM Rozszerzenie funkcji
 30 REM       SCREEN$
 40 REM ********************
 50
 60 DATA BIN 00000000
 70 DATA BIN 01111100
 80 DATA BIN 01111000
 90 DATA BIN 01111000
100 DATA BIN 01111100
110 DATA BIN 01001110
120 DATA BIN 00000111
130 DATA BIN 00000010
140
150 FOR F=0 TO 7
160    READ X
170    POKE USR "A"+F,X
180 NEXT F
190
200 LET UDG=USR "A"
210 LET CHRS=UDG-520
220 LET X=RND*31
230 LET Y=RND*20
240 PRINT AT Y,X;"A"
250 PRINT #0;" Naciskaj klawisz
e Q,A,O,P tak, aby kwadrat traf
il w strzalke."
260 LET X=16: LET Y=11
270 PRINT AT Y,X; OVER 1;"■"
280 LET XX=0: LET YY=0
290 POKE 23658,8: LET K$=INKEY$
300 IF K$="Q" AND (Y>0) THEN L
ET YY=-1
310 IF K$="A" AND (Y<20) THEN
LET YY=1
320 IF K$="O" AND (X>0) THEN L
ET XX=-1
330 IF K$="P" AND (X<31) THEN
LET XX=1
340 GO SUB 1000
350 IF SCREEN$ (Y,X)="A" THEN
GO SUB 2000: STOP
360 GO SUB 2000
370 PRINT AT Y,X; OVER 1;"■"
380 LET X=X+XX: LET Y=Y+YY
390 GO TO 270
400
1000 POKE 23607,INT (CHRS/256)
1010 POKE 23606,(CHRS-256.00001*
PEEK 23607)
1020 RETURN
1030
2000 POKE 23607,60
2010 POKE 23606,0
2020 RETURN
```

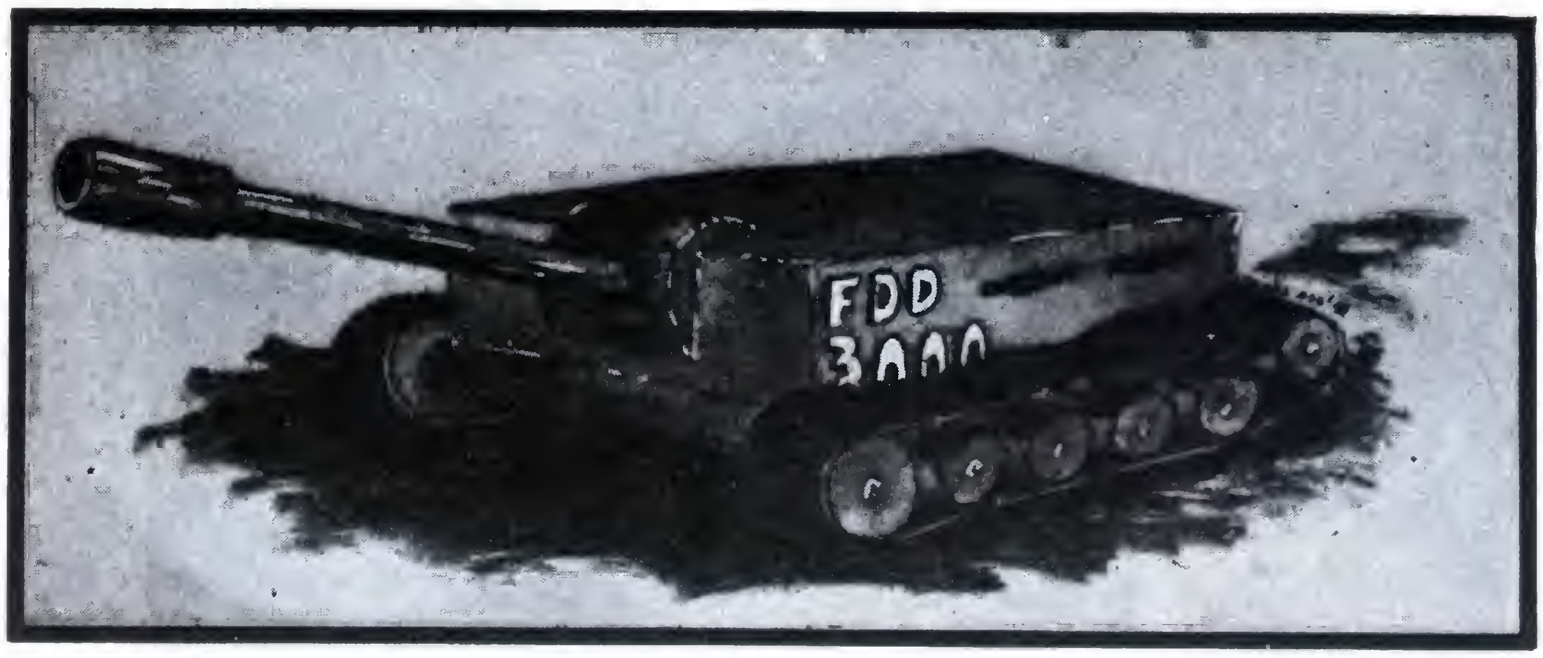
Funkcja **SCREEN$** rozpoznaje tylko znaki o kodach 32-127, tzn. tylko te, które są zdefiniowane w ROM-ie, ale co ciekawe, radzi sobie również ze znakami drukowanymi przez **INVERSE**. Aby rozpoznała ona znak o innym kodzie lub inny krój liter (character font), trzeba ustawić zmienną **CHRS** na adres, gdzie znajdują się definicje nowych znaków. Chcemy na przykład rozpoznać znak o kodzie 144, czyli pierwszy znak **UDG** ukryty pod graficzną literą „A":

najpierw definiujemy ten znak (na przykład strzałkę) w liniach 60-130 programu i wpisujemy go (w liniach 150-180) pod tę literę. Następnie adres początku jej definicji minus 520 wpisujemy do zmiennej **CHRS** (linie 200-210 i 1000-1020). Teraz możemy rozpoznać ten znak na ekranie pisząc np. **IF SCREEN$ (Y,X)="A"**, ale uwaga: piszemy tu zwykłą, dużą literą "A" (zamiast graficznego "A")! Musimy tak postąpić, gdyż funkcja **SCREEN$** nadal rozpoznaje tylko znaki o kodach 32-127, ale tym razem znak na ekranie porównywany jest z innym wzorcem: duża litera "A" ma kod 65, a każdy znak ma 8 bajtów, co razem daje 65∗8=520!

Stąd właśnie pod zwykłą dużą literą "A" można rozpoznać znak zdefiniowany pod graficzną literą "A". Pozostaje jeszcze przywrócić poprzednią wartość zmiennej **CHRS**, aby poprawnie działało polecenie PRINT (linie 2000-2020).

Powyższe uwagi można wykorzystać pisząc własne programy, i to niekoniecznie w BASIC-u. Modyfikując odpowiednio zmienną **CHRS** można wydrukować i rozpoznać na ekranie dowolny znak, pod warunkiem, że w pamięci jest jego wzorzec.

Marek Sawicki



PROSTY EDYTOR DYSKOWY

Dostęp do zawartości TOS-owej dyskietki możliwy jest na kilka różnych sposobów. Najmniej używanym na co dzień, bo i najmniej przejrzystym jest edycja sektorów.

Na naszej dyskietce jest — gwoli przypomnienia — 40 ścieżek po 16 sektorów. W każdym sektorze mieści się 256 bajtów, na ścieżce 4096 bajtów. Wygodną edycję sektorów zapewnia m.in. program narzędziowy TC_TOOLS, lecz jest długi.

Przygotowaliśmy niewielki edytor, na listingu obok. Pierwszy listing to BASIC-owa część, do wpisania i nagrania komendą **SAVE *"SECTOR"LINE 1.** Drugi listing po wpisaniu i uruchomieniu automatycznie nagra plik SECTOR.1.

Edycja sektorów polega na wczytaniu pod adres 32768 danej ścieżki i wyświetlaniu jej zawartości partiami po 256 bajtów. Poruszanie się po dysku możliwe jest za pomocą klawiszy góra/dół, zmiana zaś zawartości za pomocą klawiszy:

O (POKE) — jednej komórki
T (TEXT) — zawartości kilku bajtów

Poza tym:
M — zmiana ścieżki
S — nagranie ścieżki.

Michał Szafrański

LISTING 2

```
   1 DATA 195,139,117,195,82,119,195,171,117,0,1330
   2 DATA 128,0,0,22,0,0,245,58,61,117,631
   3 DATA 215,58,62,117,215,58,63,117,215,241,1361
   4 DATA 201,62,2,205,1,22,42,57,117,14,723
   5 DATA 16,175,50,62,117,50,63,117,6,16,672
   6 DATA 205,64,117,205,62,119,62,46,245,55,1180
   7 DATA 63,126,254,32,56,7,254,165,48,3,1008
   8 DATA 215,24,3,241,245,215,35,16,236,58,1288
   9 DATA 62,117,60,50,62,117,241,13,32,214,968
  10 DATA 201,243,62,2,205,1,22,33,0,128,897
  11 DATA 34,57,117,175,50,60,117,58,59,117,844
  12 DATA 50,50,119,60,50,49,119,50,59,117,723
  13 DATA 205,40,118,205,79,117,205,105,118,205,1397
  14 DATA 142,2,123,254,33,40,54,254,3,40,945
  15 DATA 55,254,11,40,116,254,16,32,10,22,810
  16 DATA 1,245,122,50,129,92,241,251,201,254,1586
  17 DATA 30,32,4,22,2,24,240,254,25,32,665
  18 DATA 4,22,3,24,232,254,34,32,4,22,631
  19 DATA 4,24,224,254,5,32,198,22,5,24,792
  20 DATA 216,205,52,119,251,201,205,52,119,42,1462
  21 DATA 57,117,36,58,60,117,254,15,40,9,763
  22 DATA 60,50,60,117,34,57,117,24,160,58,737
  23 DATA 59,117,254,39,40,20,60,50,59,117,815
  24 DATA 205,40,118,33,0,128,34,57,117,34,766
  25 DATA 14,119,175,50,60,117,24,131,245,50,985
  26 DATA 50,119,60,50,49,119,205,183,118,241,1194
  27 DATA 201,205,52,119,42,57,117,37,58,60,948
  28 DATA 117,254,0,40,10,61,50,60,117,34,743
  29 DATA 57,117,195,171,117,58,59,117,254,0,1145
  30 DATA 40,18,61,50,59,117,205,40,118,33,741
  31 DATA 0,143,34,57,117,62,15,50,60,117,655
  32 DATA 195,171,117,33,153,118,205,136,118,58,1304
  33 DATA 60,117,111,38,0,205,144,118,33,168,994
  34 DATA 118,205,136,118,58,59,117,111,38,0,960
  35 DATA 205,144,118,201,126,254,255,200,215,35,1753

  36 DATA 24,248,229,193,205,43,45,205,227,45,1464
  37 DATA 201,22,17,4,83,69,67,84,79,82,708
  38 DATA 58,32,32,8,8,255,22,19,4,84,522
  39 DATA 82,65,67,75,32,58,32,32,8,8,459
  40 DATA 255,195,34,119,253,229,245,253,33,0,1616
  41 DATA 0,205,8,0,241,253,225,201,205,8,1346
  42 DATA 6,205,11,6,48,251,58,2,33,183,803
  43 DATA 200,205,3,6,237,123,47,119,24,68,1032
  44 DATA 213,197,205,186,118,30,0,62,27,50,1088
  45 DATA 0,33,58,51,119,79,213,229,205,200,1187
  46 DATA 118,225,17,0,32,235,1,0,1,237,866
  47 DATA 176,235,209,123,198,7,230,15,95,32,1320
  48 DATA 222,205,3,6,193,209,201,33,0,128,1200
  49 DATA 58,49,119,71,58,50,119,87,205,222,1038
  50 DATA 118,20,122,184,200,16,247,201,237,115,1460
  51 DATA 47,119,62,2,205,1,22,205,13,119,795
  52 DATA 201,60,111,5,4,0,17,0,0,33,431
  53 DATA 200,0,205,181,3,201,229,197,213,245,1674
  54 DATA 68,77,205,43,45,205,227,45,62,32,1009
  55 DATA 215,241,209,193,225,201,243,58,59,117,1761
  56 DATA 50,222,119,60,50,221,119,205,206,119,1371
  57 DATA 251,201,253,229,245,253,33,0,0,205,1670
  58 DATA 8,0,241,253,225,201,205,8,6,205,1352
  59 DATA 11,6,48,251,58,2,33,183,200,205,997
  60 DATA 3,6,237,123,219,119,24,72,213,197,1213
  61 DATA 205,98,119,30,0,213,17,0,32,1,715
  62 DATA 0,1,237,176,209,213,229,175,205,5,1450
  63 DATA 6,62,28,50,0,33,58,223,119,79,658
  64 DATA 205,112,119,225,209,123,198,7,230,15,1443
  65 DATA 95,32,218,205,3,6,193,209,201,33,1195
  66 DATA 0,128,58,221,119,71,58,222,119,87,1083
  67 DATA 205,134,119,20,122,184,200,16,247,201,1448
  68 DATA 237,115,219,119,62,2,205,1,22,205,1187
  69 DATA 185,119,201,60,111,5,4,0,0,0,685
  70 DATA 0,0,0,0,0,0,0,0,0,0
8000 CLEAR 62999: LET a=63000
8010 FOR n=1 TO 70
8020 LET s=0
8030 FOR m=0 TO 9
8040 READ w: POKE a,w: LET s=s+w: LET a=a+1
8050 NEXT m: READ w: IF w<>s THEN  PRINT "Popraw lini
e ";n: STOP
8060 NEXT n
8070 SAVE *"sector.1"CODE 63e3,700
```

LISTING 1

```
  10 REM     1990   DR SZAFRAN
  20 POKE VAL "23658",VAL "8": BORDER NO
T PI: PAPER NOT PI: INK VAL "9": CLEAR V
AL "29999"
  30 LOAD *"SECTOR.1"CODE VAL "3E4": POK
E VAL "30011",VAL "3"
  35 PRINT AT VAL "10",VAL "4";"CHANGE D
ISK & PRESS KEY": PAUSE NOT PI: CLS
  40 POKE VAL "23681",NOT PI: RANDOMIZE
USR VAL "3E4": GO TO CODE "2"
  45 POKE VAL "23681",NOT PI: RANDOMIZE
USR VAL "30006"
  50 LET a=PEEK VAL "23681"
  60 IF a=SGN PI THEN  GO TO CODE "x"
  70 IF a=VAL "2" THEN  GO TO VAL "150"
  80 IF a=VAL "3" THEN  GO TO CODE " BRI
GHT "
  90 IF a=VAL "4" THEN  GO TO VAL "310"
 100 IF a=VAL "5" THEN  GO TO VAL "350"
 110 GO TO VAL "9999"
 120 INPUT "TRACK:"; LINE A$
 130 IF VAL A$<NOT PI OR VAL A$>CODE "'"
 THEN  GO TO CODE "x"
 140 POKE VAL "30011",VAL A$-SGN PI: GO
TO CODE "("
 150 PRINT #NOT PI;AT SGN PI,NOT PI;"SAV
E (Y/N)?"
 155 PAUSE NOT PI
 160 IF INKEY$="" THEN  GO TO VAL "160"
 170 IF INKEY$<>"Y" THEN  INPUT ;: GO TO
CODE "("
 200 INPUT ;: RANDOMIZE USR VAL "30003"
 210 GO TO CODE "-"
 220 PRINT #NOT PI;AT SGN PI,NOT PI;"LOA
D (Y/N)?"
 225 PAUSE NOT PI
 230 IF INKEY$="" THEN  GO TO VAL "230"
 240 IF INKEY$<>"Y" THEN  INPUT ;: GO TO
CODE "("
 250 INPUT "NAME:"; LINE A$
 260 IF LEN A$>VAL "11" THEN  GO TO VAL
"250"
 270 INPUT "START:"; LINE B$
 280 IF VAL B$<VAL "32768" OR VAL B$>VAL
"36864" THEN  GO TO VAL "270"
 290 LOAD *A$CODE VAL B$
 300 INPUT "              CHANGE  DISK": PAU
SE NOT PI: INPUT ;: GO TO CODE "-"
 310 INPUT "POKE ";ADR;",";WRT
 320 IF ADR>VAL "32768" AND ADR<VAL "368
64" THEN  GO TO VAL "340": IF WRT>NOT PI
 AND WRT<VAL "255" THEN  GO TO VAL "340"
 330 GO TO VAL "310"
 340 POKE ADR,WRT: GO TO CODE "-"
 350 INPUT "FROM:";ADR
 360 IF ADR<VAL "32768" OR ADR>VAL "3686
4" THEN  GO TO VAL "350"
 370 POKE VAL "23658",NOT PI: INPUT "TEX
T:"; LINE A$
 380 POKE VAL "23658",VAL "8": FOR A=SGN
 PI TO LEN A$: POKE ADR-SGN PI+A,CODE A$
(A TO A): NEXT A
 390 GO TO CODE "-"
```

JĘZYK MASZYNOWY cz. 6

W najbliższych odcinkach zajmiemy się omówieniem procedury PIXEL-ADD z ROM-u Spectrum.

Procedura ta rezyduje pod adresem 8874 (# 22AA) i ustala adres danego bajtu ekranu za pomocą podanych współrzędnych punktu. Aby jednak rozpocząć omówienie, należy przybliżyć sposób organizacji ekranu Spectrum.

Ekran podzielony został poziomo na trzy równe części: górną, środkową i dolną. Każda z nich składa się z ośmiu wierszy, zaś każdy wiersz z ośmiu linii. Daje to w sumie 192 linie i taka jest pionowa rozdzielczość ekranu.

Z kolei w poziomie każdą linię podzielono na trzydzieści dwa „odcinki" po osiem punktów (pixeli). Każdy „odcinek" reprezentowany jest w pamięci przez jeden bajt. Daje to w poziomie rozdzielczość 256 punktów (pionowych linii).

Brak podziału na ekran graficzny i tekstowy wymusza wyświetlanie liter na ekranie graficzny. Wykorzystano 3*8=24 wiersze poziome po 8 pixeli, daje to tekstową rozdzielczość 32x24.

Odwołując się do określonych części ekranu podajemy współrzędne interesującego nas punktu lub pola. Pixele ponumerowano poczynając od lewego dolnego rogu: od 0 do 255 w poziomie i od 0 do 175 w pionie. Dlaczego nie do 191? Otóż dwa dolne wiersze (16 pixeli) stanowią tzw. okno systemowe, do którego dostęp jest inny, niż do pozostałej części ekranu. W Basic-u nie ma w ogóle możliwości „sięgnięcia" tam instrukcją PLOT. Tym niemniej obszar ten jest wykorzystywany.

Dla tekstu udostępniono również tylko górne 22 wiersze znakowe i ponumerowano je (od góry) od 0 do 21, zaś znaki w poziomie od 0 do 31.

Wyświetlenie obrazu jakiejkolwiek treści to umieszczenie w pamięci ekranu takiego ciągu bajtów, by na ekranie znalazły się pixele w odpowiedniej kombinacji. Z optycznego punktu widzenia pixel „zapalony" to czarny, nie świecący punkt na ekranie, zaś pixel „zgaszony" to biały, świecący punkt. Pixelowi zapalonemu odpowiada bit 1, zgaszonemu zaś bit 0. W ten sposób, przyporządkowując kolejnym adresom w pamięci kolejne odcinki po osiem pixeli (bajty) otrzymujemy gotowy sposób sterowania obrazem.

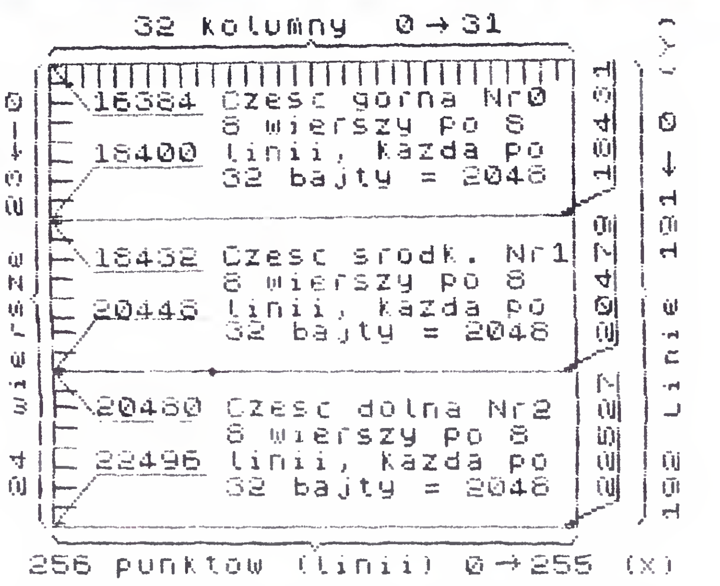
Pierwszy bajt ekranu ma adres 16384, zaś ostatni 22527. W sumie pamięć obrazu (dla pixeli) zajmuje 32*24*8=6144 bajtów. Trzy poziome części ekranu obejmują kolejno adresy: górna od 16384 do 18431, środkowa od 18432 do 20479, dolna od 20480 do 22527. Na razie wszystko jest po kolei.

W każdej części numeracja przebiega z lewa na prawo oraz z góry na dół z krokiem co osiem. Na przykład, w części pierwszej numeracja rozpoczyna się od adresu 16384 i najpierw numerowana jest pierwsza linia (nr 0) pierwszego wiersza (nr 0), następnie pierwsza linia (nr 7) drugiego wiersza (nr 1), następnie pierwsza linia (nr 15) trzeciego wiersza (nr 2) itd. Po dotarciu do ósmego wiersza (nr 7) numerowane są w ten sam sposób drugie linie kolejnych wierszy.

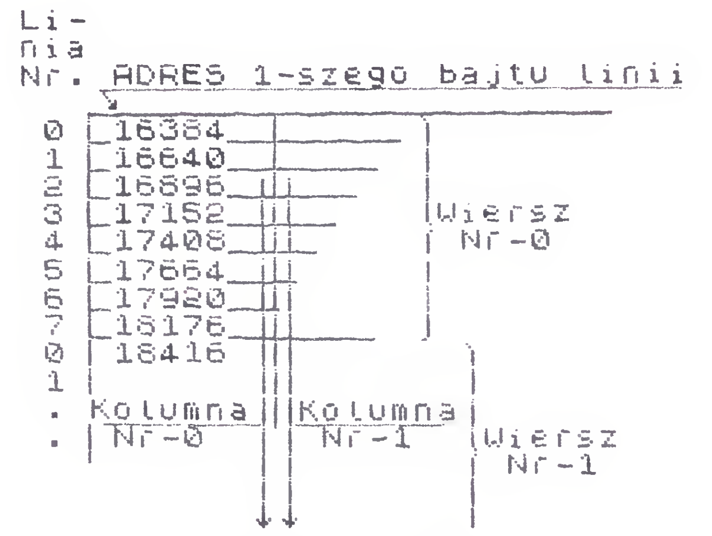
Mając dane współrzędną poziomą i pionową interesującego nas pixela na ekranie, procedura PIXEL-ADD zwraca adres bajtu obejmującego ten pixel i numer bitu w tym bajcie. W ten sposób z poziomu języka maszynowego możemy korzystając z Basicowych przyzwyczajeń obsługiwać ekran za pomocą pary współrzędnych.

Procedura PIXEL-ADD korzysta z bardzo złożonego wzoru, którego postać i analiza w następnym odcinku.

Piotr Sumara



a) Podstawowe dane oraz podział na części, wiersze, kolumny i linie.



b) Podział na linie wiersza nr "0" (pierwszy wiersz od góry ekranu) oraz adresy początkowych bajtów tych linii.

TURBO PASCAL

— ramki semigraficzne

Dzięki programowi obsługi terminala możliwe jest podłączenie komputera ZX Spectrum do stacji dysków FDD 3000 firmy Timex i praca pod kontrolą systemu operacyjnego CP/M 2.2. W wersji oferowanej przez Polbrit wspomniany program terminala ma generator znaków zawierający m.in. polskie litery.

Każdy znak pamiętany jest jako ciąg ośmiu bajtów. Nie jest to rozwiązanie ekonomiczne ze względu na zużycie pamięci, ponieważ połowa każdego bajtu jest nie wykorzystana. Niektóre znaki generatora nie są zdefiniowane. Dzięki ekstrakodom terminala generator jest dostępny z systemu CP/M i może być zmieniony.

Na listingu 1 przedstawiono plik SYMBOL.INC zawierający dwie procedury. Pierwsza z nich o nazwie **SYMBOL**, umożliwia definicję własnego znaku, a druga — **FrameChars** — definiuje semigraficzne ramki użyteczne przy tworzeniu tabelek.

Zastosowane kody ramek zgodne są ze standardem dostępnym w systemie CP/M Plus na komputerach Amstrad PCW i CPC („Bajtek" 12/88). Ze względu na ograniczony rozmiar generatora (maksymalny kod znaku — 161) nie jest możliwe zastosowanie kodów IBM.

Rys. 1 jest ilustracją działania programu FramesOnZX,. przedstawionego na listingu 2.

Jonasz Mayer

```
(**********************************************)
(*     Plik SYMBOL,INC   (C) JM 1990          *)
(* Zawiera procedure definiujaca wlasne znaki *)
(* oraz procedure definiujaca kody ramek semi- *)
(* graficznych,                               *)
(**********************************************)

procedure Symbol (CharNo,
                  rl,r2,r3,r4,r5,r6,r7,r8 : byte);
(**********************************************)
(* definiuje znak o kodzie CharNo i okreslony *)
(* przez 8 kolejnych bajtow, CharNo = [32,,161]*)
(**********************************************)
const addrGen = $FBE9;
var   address ; integer;
begin
  address := addrGen + 8 * (CharNo - 32);
  write (#27'N',
         chr (Lo (address)), chr (Hi (address)),
         chr (8), chr (0),
         chr(r1),chr(r2),chr(r3),chr(r4),
         chr(r5),chr(r6),chr(r7),chr(r8) );
end; { of Symbol }

procedure FrameChars;
(**********************************************)
(* definiuje znaki ramek                      *)
(**********************************************)
begin
  symbol (150, $00,$00,$00,$70,$70,$60,$60,$60);
  symbol (151, $60,$60,$60,$70,$70,$60,$60,$60);
  symbol (147, $60,$60,$60,$70,$70,$00,$00,$00);

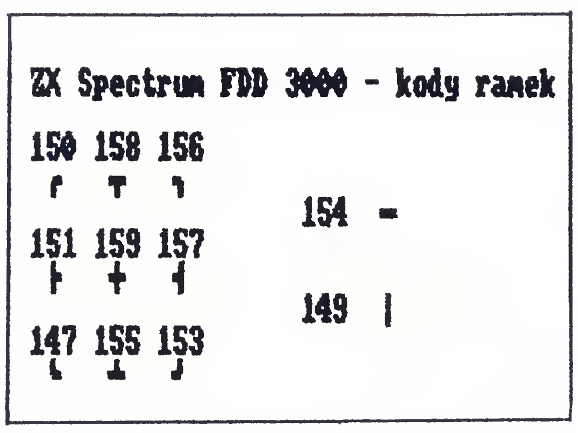
  symbol (158, $00,$00,$00,$F0,$F0,$60,$60,$60);
  symbol (159, $60,$60,$60,$F0,$F0,$60,$60,$60);
  symbol (155, $60,$60,$60,$F0,$F0,$00,$00,$00);

  symbol (156, $00,$00,$00,$E0,$E0,$60,$60,$60);
  symbol (157, $60,$60,$60,$E0,$E0,$60,$60,$60);
  symbol (153, $60,$60,$60,$E0,$E0,$00,$00,$00);

  symbol (154, $00,$00,$00,$F0,$F0,$00,$00,$00);
  symbol (149, $60,$60,$60,$60,$60,$60,$60,$60);
end; { of FrameChars }

(**********************************************)
```

Listing 1, Plik SYMBOL,INC



Rys. 1 Efekt wykonania programu FramesOnZX

```
Program FramesOnZX;
{***********************************************************************}
{                 Plik FDEMO,PAS       (C) JM 1990                      }
{ Demonstruje dzialanie procedury FrameChars definiujacej kody          }
{ ramek dla terminala Polbritu,                                         }
{***********************************************************************}

{$I SYMBOL,INC }

begin
  FrameChars;

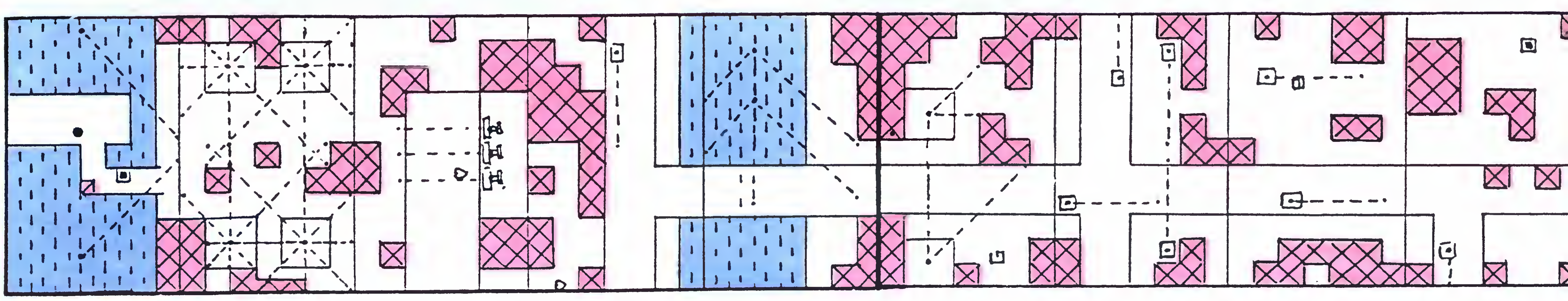
  ClrScr;
  Writeln ('ZX Spectrum FDD 3000 - kody ramek');
  writeln;
  writeln ('150 158 156');
  writeln (' '#150'  '#158'  '#156); writeln ('154':20,' '#154);

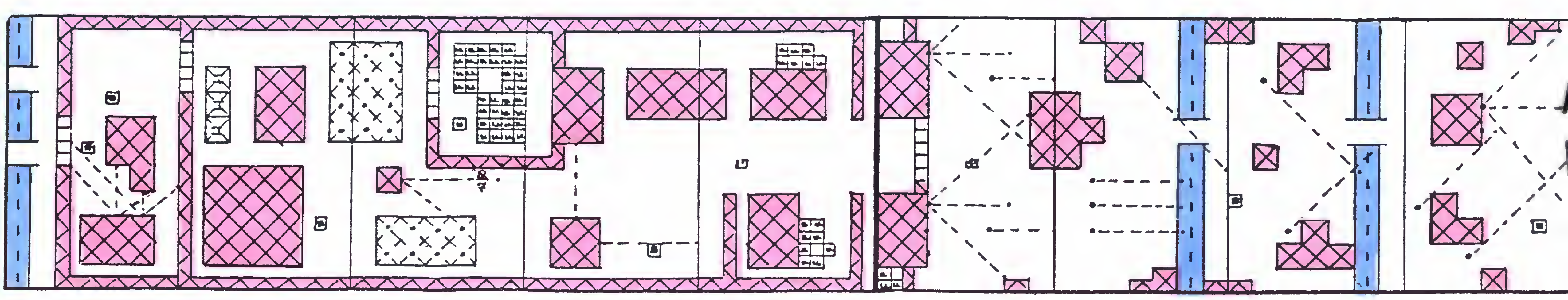
  writeln ('151 159 157');
  writeln (' '#151'  '#159'  '#157); writeln ('149':20,' '#149);

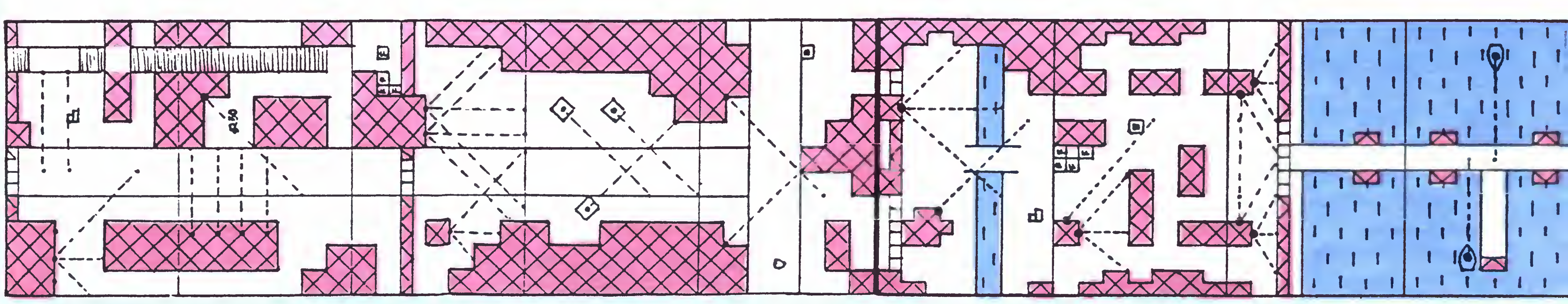
  writeln ('147 155 153');
  writeln (' '#147'  '#155'  '#153); writeln;
end,
{***********************************************************************}
```

Listing 2, Plik FDEMO,PAS









TANK COMMANDER



Dowódco! Dość już ciągłego obżerania się. Jak tak dalej pójdzie, to Twój brzuch będzie większy od najpotężniejszego czołgu. Marsz do jednostki. I to już!

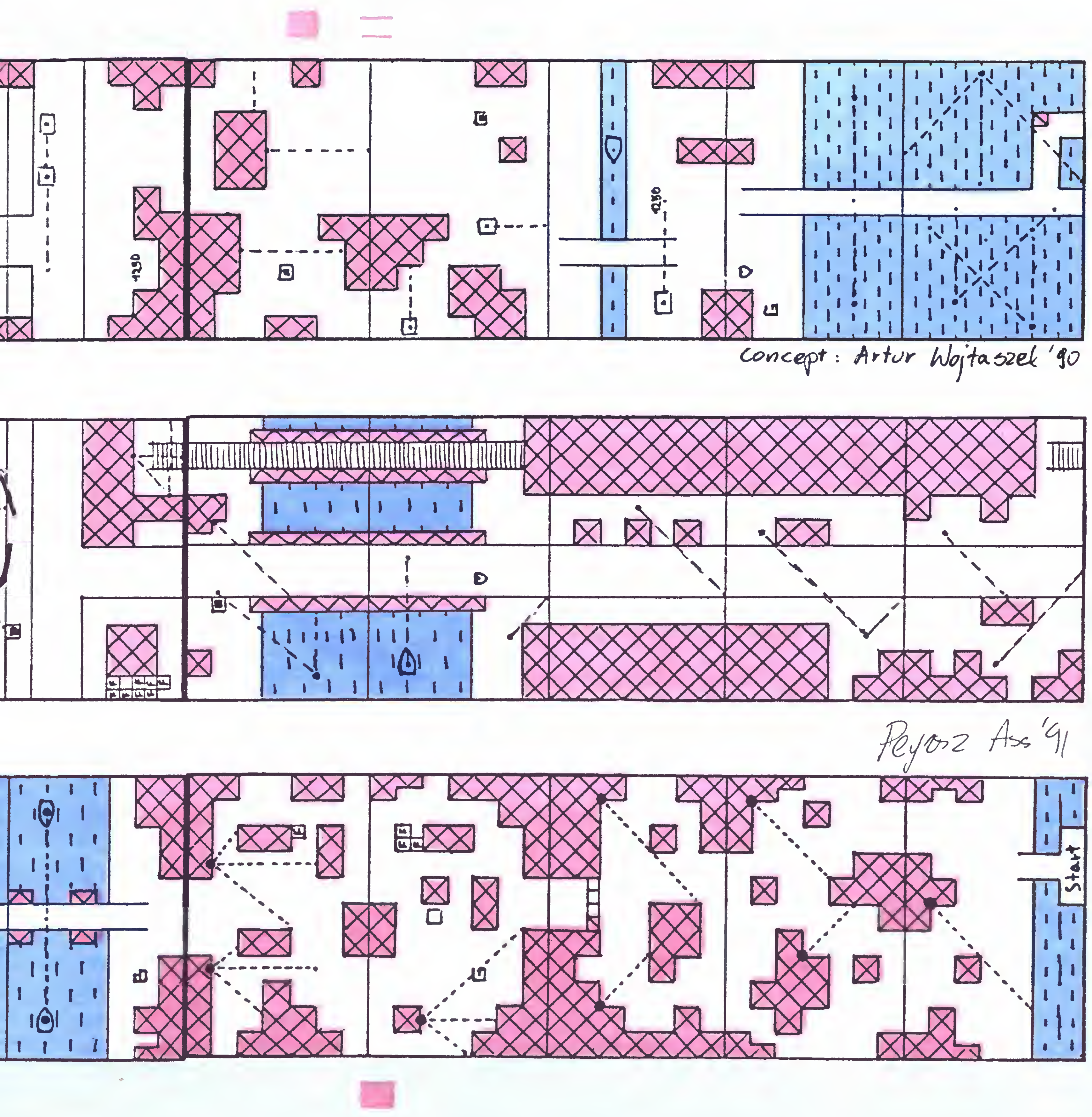
Zapraszamy do podróży w smętnym pojeździe gąsiennicowym, zwanym przez optymistów czołgiem. Działo niby jest, ale strzela jak karabin maszynowy. Trochę nieciekawie, gdy do zniszczenia są raczej cele duże i opancerzone. Możesz bić brawo autorom gry.

Uważać musisz na działka, strzelców miny. Najgorsze są te ostatnie, które kończą Twoją zabawę od razu. Działka, przy odpowiednim objechaniu są do zniszczenia. Podobnie strzelcy, których jest jednak dużo mniej.

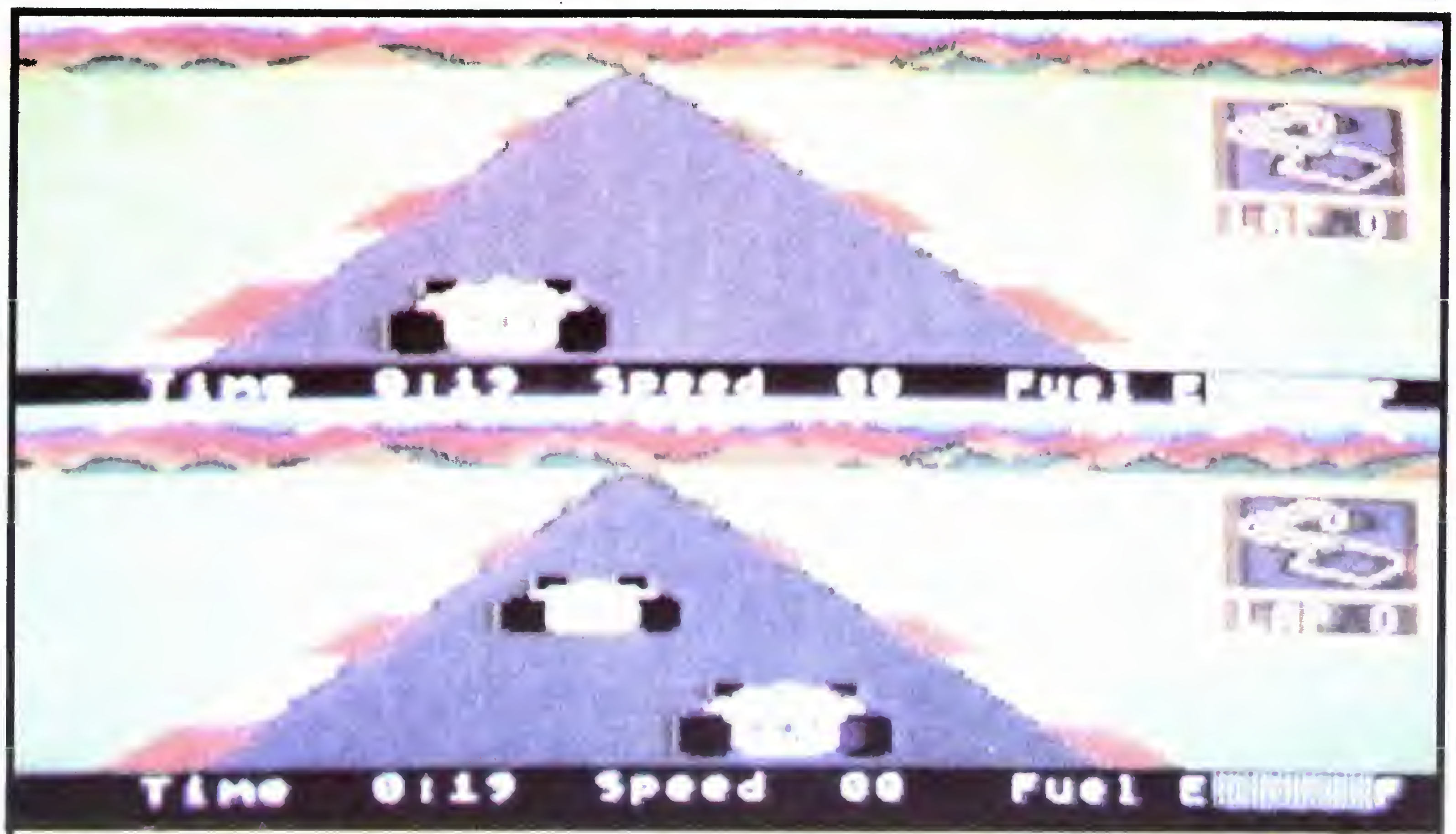
Przez wodę przedostajesz się tylko i wyłącznie mostami. Inaczej może być nieciekawie. Tory kolejowe są przejezdne w każdym miejscu.

Amunicję uzupełniasz na literach (B) bomby i (G) granaty. Zapas paliwa, podawany jest w kanistrach (F). Ekstra życie przypomina serduszko. Premie liczbowe są widoczne z daleka.

Mapę dla wersji Spectrumowskiej nadesłał Artur Wojtaszek



- Pola niedostępne
- Woda
- Pomost
- Mina
- Kanistry
- Tory kolejowe
- Extra życie
- Extra bomby
- Extra granaty
- Bonus
- Działka, strzelcy



PITSTOP

Mam radochę. Rodzice kupili mi komputer. Teraz zapomnieli o moim istnieniu i zamknęli się w sypialni. Chciałem z nimi porozmawiać, ale nie udało się. Polazłem więc na giełdę. Tam ludzi jak mrówków, a ja sam. Wydałem wszystkie pieniądze i stało się. Mam Pitstopów.

Moim chrztem bojowym było wgrywanie gry. Zjechałem magnetofon, uszy i nerwy sąsiadów. Udało się. Zabrzmiała muzyka, pojawiło się *Menu.*

Angielskie *Menu.* A ja biedny, po angielsku umiałem tylko mówić ze wspaniałym akcentem **YES** i **NO**. Więcej nie zdążyłem się nauczyć. W sumie są ciekawsze rzeczy do robienia na lekcjach angielskiego. W „kropki" jestem niepokonany.

Życie jest pełne wyrzeczeń. Zadzwoniłem do koleżanki-kujonki, bardzo miłej i co ważniejsze usłużnej. Po kilkugodzinnym wdzięczeniu się do siebie wyciągnąłem od niej wszystko, co było trzeba. Podziękowałem grzecznie z pasją odkładając słuchawkę.

Najpierw wybiera się jedną z sześciu tras istniejących w rzeczywistości na świecie. Za jednym zamachem ustalamy też liczbę okrążeń **(LAPS)** — trzy, sześć lub dziewięć. Potem migiem (nie samolotem) decydujemy się czy gramy z komputerem, czy nie: **(PLAYERS: 1—2)**. I jeszcze tylko poziom: **EASY** dla świeżego narybku, **NORMAL** dla tych normalnych oraz **ADVANCED** dla najbardziej szalonych i męskich.

W chwilę po uruchomieniu gry doznałem ekstazy. Ale to była tylko chwila. Po niej jak zwykle przyszła refleksja. Przykra refleksja.

Co my tu widziem? Ekran podzielony wzdłuż na dwie (równe!) i niezależne połowy. Oznacza to, że każdemu z rajdowców przyporządkowany jest osobny ekran. Wyświetlane na nim informacje to:

TIME — czas od strzału startera,

SPEED — prędkość maszyny (od 0 do 213),

FUEL — ilość paliwa (**E**mpty — **F**ull), gdzie **E** oznacza pusty bak a **F** pełny jak smok,

LAP — liczba przejechanych okrążeń.

Oficjalna zasada jest tylko jedna — wygrać. Nieoficjalne zaś długo by wyliczać. Można więc ścierać opony przeciwnikowi. Jeśli to sprytnie i szybko zrobisz, to wyląduje w rowie odrobinkę zszokowany. Dobrze mu tak.

Istnieje też tzw. „zasada mijanki". Polega ona na tym, że trzymasz się blisko przeciwnika do chwili wjechania w ostry zakręt. Wtedy momentalnie przyspieszasz i trzymasz się „po zewnętrznej". Mimo że po chwili wpadniesz na pobocze ścierając trochę opony, to przy umiejętnym zwrocie znajdziesz się przed przeciwnikiem.

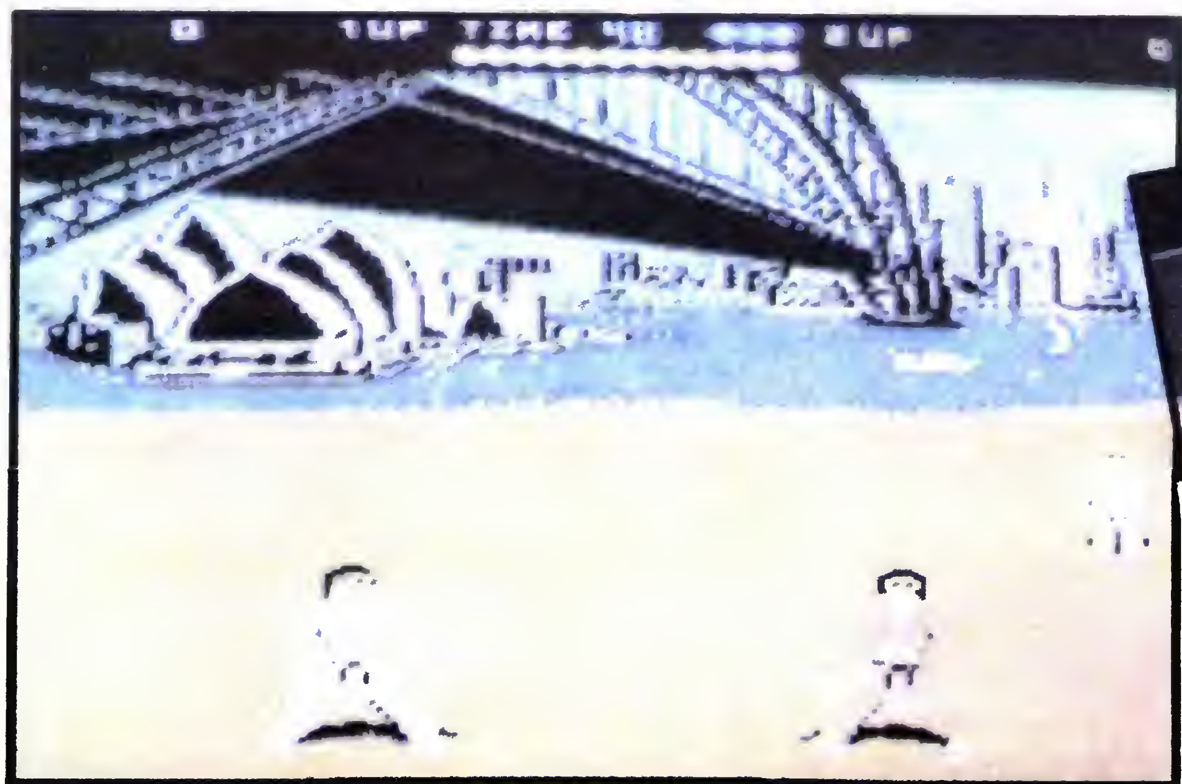
W pitstopach naprawiasz zjechane opony i uzupełniasz zapas paliwa. Patrzysz też ze zgrozą na uciekający czas. Ręce Ci się trzęsą ze zdenerwowania, nie trafiasz wężem do dziury paliwowej, przygniatasz sobie nogę wypuszczoną z rąk oponą. Moje condolencje.

Jadziem Panowie, jak mawia Pan Leon. Mądry z niego człowiek, ale nie wie, co to znaczy wcisnąć pedał do oporu. Wy, po przeczytaniu tego opisu od deski do deski, staniecie się naprawdę dobrzy.

A ja już muszę wstać...

Lavender Blue

Firma: Epyx
Rok produkcji: 1984/86
Komputer: Spectrum, Commodore, Amstrad, Atari, IBM PC





WORLD KARATE CHAMPIONSHIP

Czy pamiętasz może film „Nieśmiertelny" (nie mylić z „Nietykalnymi")? Jego fabuła była równie ambitna co nieskomplikowana: z kilkunastu Nieśmiertelnych mógł pozostać tylko jeden, wykończywszy przedtem resztę przy pomocy chirurgicznie precyzyjnych cięć dwuręcznym mieczem na wysokości szyi. Gra „World Karate Championship" opiera się na podobnej zasadzie — jedynie najlepszy z karateków uchowa się przy życiu, wszyscy pozostali odejdą z tego świata. Kto tego dokona, zostanie (Wnimanje! Wszyscy proszeni są o powstanie i przyjęcie postawy zasadniczej.) MISTRZEM!

Droga do szczęścia, ja zwykle długa i ciernista. Napotkasz wielu wojowników: od narybku ledwie z białym czy żółtym pasem, poprzez profesjonalnych morderców z pasem zielonym, purpurowym lub brązowym, po nielicznych i owianych mgłą tajemnicy wirtuozów walki z pasem czarnym. Jeśli twoja wola zwycięstwa jest naprawdę niezłomna — masz szansę. Jeśli nie — w promieniu 20 kilometrów będzie słychać, jak umierasz.

Próbuj więc, choć prawdę mówiąc nie wierzę, by ci się udało. Musisz działać w pełni odruchowo, nie znajdziesz czasu na myślenie. W tej grze pięść jest szybsza od myśli. Wachlarz ciosów jest niezwykle szeroki, co umożliwia przyjmowanie różnych strategii walki — począwszy od delikatnych kopnięć z wyskoku w twarz aż do brutalnych uderzeń we wrażliwe miejsca na ciele przeciwnika. Posługuj się przy tym niezawodnym wskaźnikiem twojej precyzji — jeśli np. wróg zawyje nieludzkim głosem i pełen cierpienia zwinie się w pozycji emorionalnej na ziemi, możesz być pewien, że trafiłeś idealnie w żołądek.

Co jakiś czas zostajesz poddany ważnej próbie. Stoisz przed stosem cegieł i musisz rozbić uderzeniem z głowy (z dyni, z bańki, z czachy) jak najwięcej z nich, co jest odpowiednio punktowane. Stanowi to jawne zaprzeczenie poglądu, jakoby to walki Wschodu nie wymagały pracy głową. Kiedy rozbijesz 20 cegieł w drobny

S.O.S.

Spiętrzenie SOS-ów w redakcji powoduje duże opóźnienia w druku Waszych wołań o pomoc. Zmieniamy nieco formułę rubryki SOS i po raz ostatni prezentujemy ją w „Bajtku". Od następnego numeru ulega też zmianie „Co jest grane", a SOS ujrzycie tylko w Top Secret.

Grzegorz Tłusty
ul. Wyspiańskiego 7/78, 39-400 Tarnobrzeg
IBM. Szuka gier z opisami.
Marcin Skowroński
ul. Białostocka 9, 05-840 Brwinów
Amstrad 464
Ma 250 gier, chce wymienić.
Jarosław Warylewski
ul. Zbożowa 13/25, Gdynia
Szuka pozbawionych błędów wersji QUESTRONA, ULTIMA IV i dalszych dysków ALTERNATE REALITY; w zamian wiele innych.
Tomasz Wickel
ul. Kopernika 2/3, 38-200 Jasło
Atari 65 XE
20 dowolnie wybranych gier da za opis do ZORRO II.
Paweł Bartoń
ul. Miła 8/37, 97-500 Radomsko
Atari 800 XL z magnetofonem
Wiele gier w zamian za inne
Paweł Adamiuk
ul. Barlickiego 3b/4, 66-620 Gubin
Atari 800 XE
Wymieni nieśmiertelności do 35 gier na programy BARBARIAN, BMX Simulator+.

Przemek Michalak
ul. Narcyzowa 1, 89-100 Występ, gm. Nakło n/Notecią
Różne gry i opisy w zamian za RAID OVER MOSCOW, BARBARIAN.
Filip Woźniak
ul. Paderewskiego 9/71, 63-101 Śrem
Atari 65 XE ze stacją
Jakie kody należy podać po wybraniu opcji w grze SOLO FLIGHT? W zamian różne gry.
Cezary Miłoszewski
ul. Daszyńskiego 96/19, 06-100 Pułtusk
Atari 65 XE. Wymieni gry.
Jarosław Krupa
ul. Sasanki 92, Rzeszów
Atari XL/XE z magnetofonem
Ma dużo fajnych gier
Szuka gry SPY vs SPY IV i informacji, jak zniszczyć toster w CHIMERZE.
Mariusz Danilewicz
ul. Jasna 28 a, 11-600 Węgorzewo
Atari 65 XE z magnetofonem
Wymieni gry.
Kuba Kiczmachowski
ul. Traugutta 2/2, 11-500 Giżycko
Atari 130 XE
Odstąpi UNIVERSAL HERO za grę GAUNTLET; nie wie, o co chodzi w grze SPELLBOUND.
Adam Chmielewski
ul. Broniewskiego 20/9, 01-780 Warszawa
Atari 130 XE
Pyta, jak zabić smoka w DRACONUS oraz szuka opisu do QUEST FOR TIRES
Da 20 wybranych gier.
Marcin Wojnarski
ul. Bulwary Słowackiego 34, 34-500 Zakopane
Atari XE
Szuka wielu gier w wersji kasetowej.
Paweł Jagiełło
ul. Mickiewicza 6/8, 06-406 Opinogóra
Atari XL/XE
Proponuje różne gry i programy z opisami
Szuka RAMBO, COMMANDO, TETRIS oraz S.A.M.
Grzegorz Puzio
ul. Okulickiego 8/1, 37-450 Stalowa Wola
Atari 130 XE. Wymieni gry i opisy.
Bartłomiej Stachyra
ul. Herbsta 1/62, 02-784 Warszawa
Atari 65 XE z magnetofonem
Wymieni gry, mapy i informacje.
Mariusz Kinicki
ul. Kolejowa 17/6, 11-015 Olsztynek
Atari 65 XE
Wymieni gry, opisy, nieśmiertelności
Marcin Krasuski
ul. Chrobrego 15/83, 08-110 Siedlce
Atari 65 XE ze stacją
Szuka OPERATION WOLF, STREET FIGHTER i PLATOON.
Bartosz Urbański
ul. H. Sawickiej 1/32, Kalisz
ZX Spectrum
Ciekawe gry, m.in. HOT SHOT w zamian za inne.
Roman Masztalerz
ul. Górna 28, 61-330 Poznań Daszewice
ZX Spectrum
Nie wie, jak usunąć toster z kwadratu G8 i G4 w grze Chimera. Sztuka też kilku opisów.
Jurand Dunaszewski
ul. Leśna 2, 40-624 Katowice — Ochojec
Spectrum 48
Prosi o kod do NAVY MOVES II; w zamian nieśmiertelności.
Piotr Zalewski
ul. Anielewicza 24 a/1, 01-052 Warszawa, tel. 38-70-80
Commodore 16
Prosi o kontakt posiadaczy takiego komputera; ma kłopoty z dostępem do gier.
Wojciech Woźniak
ul. Piastowska 90 d/61
Amstrad 464 i 6128
Szuka kilku gier oraz opisu do ROBIN OF SHERWOOD.
Rafał Jaczyński
ul. Kasprowicza 76/42, 01-949 Warszawa
Atari 65 XE z magnetofonem
Prosi o grę GAUNTLET oraz opisy.
Wojciech Pfitzner
os. Dywizjonu (303) 46/10, Nowa Huta, 31-875 Kraków
Atari 65 XE. Sztuka opisów, odstąpi inne.
Maciej Widurski
ul. Lublańska 20/72, Kraków
Atari 65 XE ze stacją
Zapłaci lub da inne za: GRYZOR, HOBBIT, THE TRAIN.
Marcin Bohdziul
ul. Koszarowa 18/12, Szczecin-Dąbie, tel. 601-453
Commodore
Wiele gier w zamian za KARNOV i LORD OF THE RINGS
Jaromir Król
Os. Piastów 8/50, 31-623 Kraków
C-64
Szuka map i gier na dysku lub kasecie.
Paweł Petasz jr.
P.O. Box 68, 82-300 Elbląg
Amiga
Kupi lub zamieni TECHNOCOP, LAST NINJA, CRAZY CARS.
Adam Serafin
ul. Małachowskiego 12 b/8, 64-800 Chodzież
Opis gry NORTH and SOUTH za informację, jak skończyć SHADOW OF THE BEAST II.
Wojciech Gieniec
ul. Szeroka 20, 33-395 Chełmiec, woj. nowosądeckie
IBM PC ze stacją
Szuka różnych gier i informacji o grach.
Daniel Bieniacz
ul. Broniewskiego 24/61, tel. 307-89, 35-206 Rzeszów Amstrad
Odda kilka gier w zamian za GREEEN BERET.
Paweł Osowski
ul. Józefowska 17, 05-860 Płochocin
ZX Spectrum z magnetofonem
Wiele opisów i gier w zamian za HOBBIT, RIVER RAID.
Adam Jasiorowski
ul. Włościańska 16/12, 01-710 Warszawa
Nawiąże kontakt z osobami znającymi Assemblera ZX Spectrum i szuka ułatwień do gry PLATOON.
Ernest Nowicki
ul. Generała Jóźwiaka 38/21, 00-500 Strzelce Krajjeńskie
C-64 z magnetofonem
Ma wiele gier. Chce się wymienić.
Michał Miedziński
ul. Kadłubka 44/7, 93-248 Łódź
C-64 ze stacją
Prosi o opisy do BLACK LAMP I GUN SHIP
Adam Kuczyński
ul. Mickiewicza 6/4, 66-600 Krosno Odrzańskie
C-64
Ma wiele gier, w tym nowości
Szuka WONDER BOY
Radosław Stefański
ul. Michałowskiego 26/71, 42-200 Częstochowa
C-64 z magnetofonem. Ma różne gry. Pragnie wymiany.
Łukasz Mroczka
ul. Schillera 4/32, 42-200 Częstochowa
Commodore. Proponuje wymianę gier.
Piotr Janasz
ul. Sambora 7, 80-381 Gdańsk-Oliwa
C-64
Nawiąże kontakt z posiadaczami C-64, wymieni gry.
Rafał Mazur
ul. Porębskiego 2 a/11, 81-185 Gdynia
C-64
Nagra ciekawe gry w zamian za inne
Marcin Aumiler
ul. Czarnkowska 23/9, 64-610 Rogoźno
Atari 800 XE i magnetofon
Ma wiele gier, programy graficzne i muzyczne. Prosi o grę GAUNTLET.
Jarosław Warylewski, lat 31
ul. Zbożowa 13/25, Gdynia
Atari 65, stacja dysków i magnetofon
Nawiąże kontakt celem wymiany oprogramowania i doświadczeń.
Artur Niździński
ul. 22 Lipca 13 a, 87-605 Huchowo, woj. włocławskie
Atari 65XE, magnetofon XC-12, system TURBO 2001, dwa joysticki. Oferuje 110 gier
Szuka wielu gier — po skopiowaniu odeśle.
Konrad Kowalewski
ul. 22-go Lipca 62/3, 72-010 Police, woj. szczecińskie
Atari 65 XE z magnetofonem Atari XCA12
Różne gry nagrane na kasecie w zamian za opis gry JUMBO JET.
Krzysztof Mazur
ul. Trybunalska 4/7, 58-100 Świdnica
Atari 65 XE
Odstąpi mapę do gry MISJA i opis do gry BASIL
Szuka dokładnych opisów do gier SPELLBOUND i BLACK LAMP.
Jarosław Butrym
ul. Czereśniowa 5 c/5, 66-400 Gorzów Wielkopolski
Atari 65 XE z magnetofonem
Różne gry i opisy. Szuka kilku gier.
Marcin Ociepa
ul. Handlowa 22, 42-262 Poczesna, woj. częstochowskie
Atari 65 XE
Odda wiele gier za pomoc w uzyskaniu innych.
Piotr Dzwonek
ul. Gałczyńskiego 5/25, 25-409 Kielce
C-64. Proponuje wymianę opisów do gier.
Jacek Trębacz
ul. Grabowa 41, 43-450 Ustroń, woj. bielskie
C-64
Oferuje 10 gier
Prosi o pomoc w dostarczeniu gry BREAK THRU i uruchomieniu F-18 HORNET.
Piotr Lewandowski
ul. Malborska 43/10, 82-300 Elbląg
C-64 z magnetofonem i stacją
Ma różne gry. Szuka gier oraz LOGO.
Andrzej Maciołek
36-071 Trzciana 352, woj. rzeszowskie
C-64 z magnetofonem
Szuka gier i pomocy w uruchomieniu.
Piotr Kobiela
ul. Beskidzka 27/6, 44-335 Jastrzębie Zdrój
C-64
Ma 800 programów. Oferuje 50 za inne.
Arkadiusz Grochowski
ul. Fr. Joliot Curie 19 a/14, 02-646 Warszawa
C-128D
Ma różne opisy i POKE' i. Szuka opisów.
Łukasz Rubka
ul. Hubala 1/91, 43-100 Tychy
C-64. Szuka wielu gier.
Jacek Wanic
ul. Broniewskiego 97/5, 01-876 Warszawa, tel. 35-37-69
C-64 ze stacją
Odstąpi różne gry z opisami
Szuka kilku gier.
Tomasz Zarzycki
59-255 Tomaszów Bol. 130, woj. legnickie
C-64
Zapłaci za interesujące go gry, literaturę i oprogramowanie lub wymieni na inne.
Mariusz Święcicki
ul. Słonecznikowa 35/28, 15-001 Białystok
C-64
Za jedną z poszukiwanych gier da dwie w wersji magnetofonowej.
Patryk Kowalski
Al. Zjednoczenia 3, Zielona Góra
C-64 ze stacją dysków
Udostępni bardzo dokładny opis do ZORRO
Prosi o grę THE LAST NINJA.
Marcin Kadłubowski
ul. Husarska 16/3, 05-120 Legionowo
C-64
Nie wie, jak uruchomić grę HEMAN INTRO.
Robert Nowakowski
ul. Mickiewicza 4/12, 82-200 Malbork
Commodore+4
Szuka odpowiedzi na wiele pytań dotyczących tego komputera.
Jakub Miła
ul. 27-go Stycznia 137/62, 41-310 Dabrowa Górnicza
C-128 (zazwyczaj gra w trybie 64)
Szuka opisów do różnych gier.
Paweł Gaudyn
Os. Kaziemierzowskie 9/24, 31-840 Kraków
C-64
Ma wiele wątpliwości co do gier na ten komputer.
Marcin Puchacz
ul. Obrońców Modlina 367/5, Nowy Dwór Mazowiecki 5
Atari 65 XE, 800 XE
Prosi o pożyczenie kaset z grami SABOTEUR I,II, BATMAN.
Dariusz Gajewski
ul. Piastowska 35/1, 58-330 Jedlina Zdrój
Atari 65 XE. Wymieni gry i opisy.
Wojciech Łabęda
ul. Piastów 4 d/34, 62-300 Września
Atari 130 XE. Wymieni gry.
Mateusz Świątecki
ul. Burskiego 16/5, 10-686 Olsztyn
Atari 65 XE i magnetofon
Prosi o opisy i nieśmiertelności na Atari.
Maciej Zieliński
ul. B. Prusa 2/35, 37-310 Nowa Sarzyna, woj. rzeszowskie
Atari z magnetofonem. Wymieni gry.
Adam Biedrzycki
ul. Żeromskiego 2, 11-600 Węgorzewo
Atari 800 XL z magnetofonem
Wymieni gry.
Piotr Stolc
ul. Wojewódzka 13/1, 81-437 Gdynia, tel. 22-02-02
C-64 z magnetofonem
Chce wymieniać gry i rozmawiać o programach.
Dominik Gajerski
ul. Banacha 12/5, 02-097 Warszawa
C-64. Nie wie, o co chodzi w kilku grach.
Patryk Wasiak
ul. Szpitalna 3, 24-200 Bełżyce
C-64
Szuka gier, ma opisy i nieśmiertelności.
Marcin Kurdziel
ul. Wyspiańskiego 17/94, 39-400 Tarnobrzeg
Szuka Bajtków 1, 5-6, 7, 8 z 86 r., 9 i 10 z 87 r., 2 i 6 z 88 r., a także opisów.

---

mak, nie przyniesie ci ujmy na honorze prośba o zimny kompres (na czoło i na zasilacz).

To jednak nie koniec niespodzianek. W pewnym momencie (z reguły kiedy idzie Ci najlepiej) duchy pokonanych podczas turnieju zawodników materializują się w postaci sztyletów, włóczni i innego paskudztwa latającego ze sporą prędkością w różne strony. Potrzebny jest niezły refleks, o ile nie chcesz, by któryś z nich zmaterializował ci się w plecach.

Jednym słowem, „World Karate" to gra dla twardzieli. Wymagana jest lotność umysłu i szybkość działania. Niezbędne okazuje się całkowite zespolenie ducha i umysłu z ciałem (tu pomóc może zarówno 10 lat medytacji, jak i butelka wina). Wreszcie nikogo razić nie może przedśmiertne rzężenie przeciwnika w agonii.

Mimo iż koncepcja programu w pewnym sensie oparta jest, jak już wspomniałem, na filmie „Nieśmiertelni", trochę jednak brakuje tej nieśmiertelności. Tylko nasi stali Czytelnicy pamiętają, że stosowne zaklęcie znajduje się w „Bajtku" 6/88 na stronie ósmej. Ale o tym sza: To wolno wiedzieć tylko wybranym.

BÁNZÁI! i...

Firma: Epyx
Rok produkcji: 1987
Komputer: Atari

Master

PRAWIE ARKA-NOID

czyli o uniwersalnym schemacie gry zręcznościowej

Wszyscy czytelnicy znają, jak sądzę, scenariusz gry *ARKANOID.* Zamysł tej gry jest dziecinnie prosty i wymyślony już u zarania mikrokomputeryzacji.

Niedawno dopiero został wyposażony w odpowiednią grafikę i oprawę muzyczną stając się w ten sposób grą o dużej atrakcyjności. Motywem, na którym opiera się *ARKANOID* jest odbijająca się piłka. Chciałbym zwrócić uwagę czytelników na ten motyw i omówić go na tyle dokładnie, by możliwe stało się napisanie własnej gry na nim opartej. Na jednym motywie np. *Arkanoida* można napisać wiele różnych gier. Porównałbym to do komponowania różnych utworów muzycznych opartych na tej samej, wpadającej w ucho melodii. Przykładem takich różnych „kompozycji" niech będą programy *Tertis* i *BLOCKOUT* oraz motyw spadających klocków. Kilka wariacji na temat latającej piłki przedstawię w kolejnych artykułach.

Pojęcia, które uległy w treści artykułu i programu dużym uproszczeniom oznaczyłem gwiazdką (* i **) i odsyłam do lektury w ramce 1 i 2.

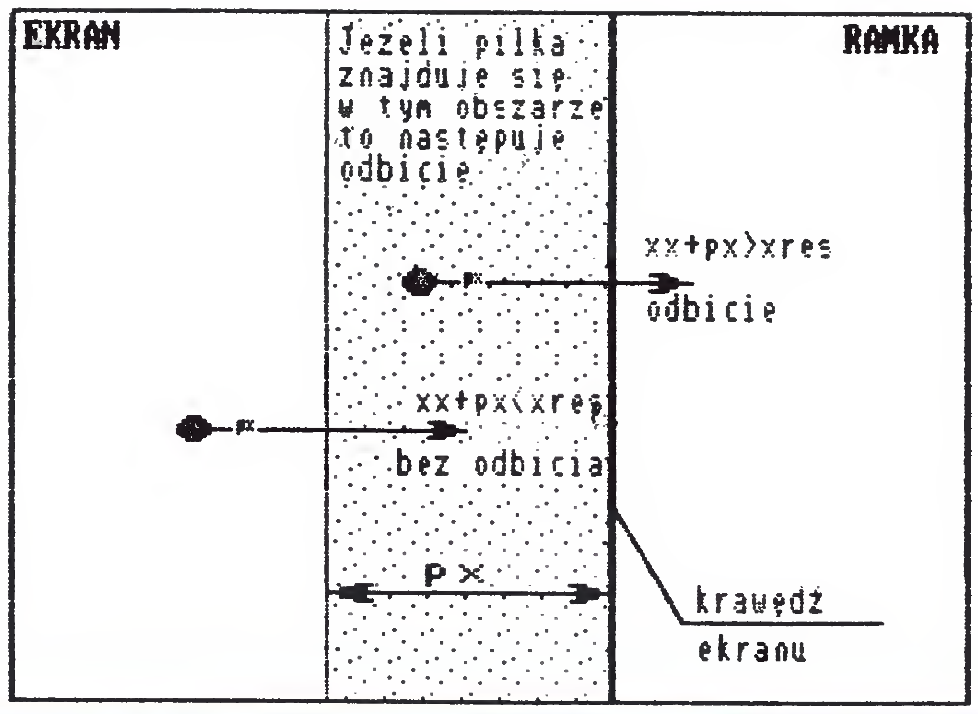
MOTYW GRY

Spróbujmy na początek zastanowić się, które cechy Arkanoida wypływają z motywu, a które są dodatkiem charakterystycznym dla tej gry. Na pewno nieistotny będzie kolor w grafice, wielkość piłki, jak i muzyczka i napisy na początku gry. Najważniejsze jest stwierdzenie, że medium motywu jest poruszająca się piłka. Dodatkowo piłka ta odbija się od napotkanych przeszkód. W dwóch ostatnich zdaniach zamyka się opis motywu gry. Zdziwić to może niektórych, ale tak bywa, iż z prostych motywów powstają często gry-bestsellery. Mamy więc szansę na rozbudowanie motywu po swojemu i nowy szlagier wśród gier.

REALIZACJA PROGRAMOWA

Poniżej przedstawiony jest program realizujący założenia omawianego motywu. By łatwiej było śledzić algorytm napisany w BASIC'u opatrzyłem go (zbyt może) licznymi komentarzami. Nie bałem się również stosować instrukcji GOTO, która (stosowana rozważnie) nie pogarsza czytelności programu, a pozwala uniknąć sztucznych rozwiązań.

Rysunek 1.




```
5 DEFINT r
10 resx=639                              ' pozioma rozdzielczość komputera
20 resy=399                              ' pionowa rozdzielczość komputera
30 MODE 1                                ' grafika o proporcjonalnych pikselach
40 xx=resx/2                             ' poziome położenie piłki
50 yy=resy/2                             ' pionowe     -      -
60 px=2                                  ' prędkość pozioma
70 py=2                                  '    -      pionowa
80 sx=xx                                 ' poprzednie położenie poziome
90 sy=yy                                 '      -          -      pionowe
500 REM -----------------------------------------------------------------
510 WHILE 1                              ' pętla nieskończona
520      xx=xx+px                        ' obliczenie nowego położenia poz.
530      yy=yy+py                        '      -        -        -     pion
540      REM -----------------------
550      IF (xx+px>resx) THEN px=-px     ' odbicie od prawej krawędzi
560      IF (xx+px<0) THEN px=-px        '    -    od lewej krawędzi
570      IF (yy+py>resy) THEN py=-py     '    -    od górnej krawędzi
580      IF (yy+py<0) THEN py=-py        '    -    od dolnej krawędzi
720      REM -----------------------
730      PLOT xx,yy,1                    ' zapalenie punktu - pilki
750      PLOT sx,sy,0                    ' zgaszenie poprzedniego punktu
760      sx=xx: sy=yy                    ' zapamiętanie poł. nowego punktu
770      REM -----------------------
780 WEND
```

Program ten nie jest duży, a i funkcje, które spełnia nie są skomplikowane. Program po zainicjowaniu zmiennych wpada w nieskończoną pętlę zrealizowaną za pomocą instrukcji WHILE-WEND. Być może przejrzyściej byłoby zastosować podprogram lecz ze względu na konieczną szybkość działania programu zaoszczędzimy na skokach do procedury. Warunki w jakich następuje odbicie od krawędzi wyjaśnia rys. 1.

RAMKA 1. KINEMATYKA PUNKTU

Analogia między programem a rzeczywistością występuje nie tylko w działaniu. Wielkości fizyczne mają swoje odpowiedniki wśród zmiennych. Skąd się zatem wzięły wzory, z których korzysta program nie jest oczywiste. W kinematyce obowiązują następujące wzory:

$S=S_o+Vt$ (I)

gdzie: S — droga (położenie);
S_o — położenie początkowe;
V — prędkość;
t — czas;

$V=V_o+½\cdot at$ (II)

gdzie: V_o — prędkość początkowa;
a — przyspieszenie;

Wzory te ulegają uproszczeniu po przyjęciu pewnych założeń, a mianowicie: iloczyn $V\cdot t$ oznacza drogę przebytą w czasie t z prędkością V. Przyjmując długość czasu $t=1$ będzie to „droga jednostkowa". Zatem zmienne *px* i *py* nazywane w programie prędkościami są w gruncie rzeczy długościami drogi przebywanej przez piłkę w jednej pętli. Na podobnej zasadzie funkcjonują zmienne: *accel* i *gravit.* Jak powstały uproszczenia wzorów kinematycznych przedstawiają poniższe równania:

(I)

$S=S_o+V\cdot 1$
$S=S_o+V$

xx=xx+px
yy=yy+py

(II)

$V=V_o+½\cdot a\cdot 1^2$
$V=V_o+½\cdot a$

px=px+accel
py=py+accel

Jedno wektorowe równanie ruchu daje dwa równania skalarne we współrzędnych kartezjańskich.

ROZBUDOWA MOTYWU

Przystąpimy teraz do rozbudowy motywu w kierunku gry. Przede wszystkim dodajmy sterowanie przyspieszeniem* piłki z klawiatury:

```
5 DEFINT r,k
100 accel=0.1                          ' przyspieszenie rakiety
130 keyup=240:    keydown=241          ' kody klawiszy kursorów
140 keyright=243: keyleft=242          '  -      -      -
590     REM ------------------------
600     a$=INKEY$                      ' pobranie znaku naciśn. klawisza
610     IF a$="" THEN GOTO 670         ' spr. czy nie pusty
620     ch=ASC(a$)                     ' zamiana na kod litery
630     IF ch=keyup THEN py=py+accel
640     IF ch=keydown THEN py=py-accel
650     IF ch=keyright THEN px=px+accel
660     IF ch=keyleft THEN px=px-accel
670     REM ------------------------
```

Realizacja kontroli klawiatury jest dziwna ze względu na zatrzymanie programu z błędem w razie podania do instrukcji ASC pustego ciągu. Wartość przyspieszenia* zawarta jest w zmiennej *accel* (ang. acceleration — przyspieszenie). Po tym dodatku zmienił się charakter naszego programu. Bezładnie odbijająca się piłka zmieniła się w sterowaną rakietę. Dodajmy więc opór powietrza (dla uproszczenia liniowy)**:

```
110 air=0.999                          ' opór powietrza
680     px=px*air                      ' poziomy opór powietrza
690     py=py*air                      ' pionowy  -     -
```

Skoro jest rakieta to powinna być i grawitacja*:

```
120 gravit=-0.02                       ' wielkość oddziaływania grawit.
700     REM ------------------------
710     py=py+gravit                   ' wpływ grawitacji
```

Warto zwrócić uwagę na zależność ruchu rakiety od dobrania trzech czynników: przyspieszenia* (zmienna accel od 0.02 do 0.5), oporu powietrza** (zmienna *air* od 0.98 do 0.999) i grawitacji* (zmienna *gravit* od -0.1 do 0).

Dla poprawienia efektu estetycznego można dodać linię:

```
740     IF (INT((sx+.5)/2)=INT((xx+.5)/2)) AND
        (INT((sy+.5)/2)=INT((yy+.5)/2)) THEN GOTO 770
```

Ze względu na łatwość wprowadzenia tematu przedstawiłem program w języku BASIC. Z całą jednak pewnością BASIC nie dorównuje przejrzystością Pascalowi. Poza tym kompilacja nadaje programom Pascalowym atut szybkości. Dlatego zamieszczam na oddzielnym wydruku pełny program w *Turbo Pascalu* i obiecuję, że następne programy w tej serii napiszę właśnie w Pascal'u. W następnym numerze ciąg dalszy rozważań nad latającą piłką.

Marcin Skóra

RAMKA 2. OPORY RUCHU

Na poruszający się punkt działa w naszym programie opór ośrodka, w którym punkt ten się porusza. W rzeczywistości opór powietrza rośnie proporcjonalnie do drugiej potęgi prędkości i wyraża się wzorem:

$$P_x = c_x \cdot \frac{g \cdot v^2}{2} \cdot S$$

gdzie: P_x — opory ruchu;
g — gęstość ośrodka;
v — prędkość obiektu;
S — przekrój powierzchni czołowej;
c_x — współczynnik zależny od kształtu obiektu (opływowości)

W naszym programie realizacja kwadratowego oporu ośrodka jest niepotrzebna i wystarczy zwykły „hamulec" zmniejszający cały czas prędkość w sposób liniowy tzn. proporcjonalny do pierwszej potęgi prędkości. Proporcję zmniejszenia prędkości wyraża zmienna *air*.



```
 1 :  {
 2 :   Pilka v 1.1
 3 :   SKOGOOT SOFTWARE 90
 4 :   Warszawa pazdziernik 1990
 5 :   }
 6 :   program pilka;
 7 :   { ****************************************************************
 8 :     Program symulujacy poruszajacy sie obiekt ze wzgledu na
 9 :     polozenie (odbicia), predkosc, przyspieszenie (klawiatura)
10 :    opor powietrza i grawitacje.
11 :    ****************************************************************
12 :  }
13 :  const Copyright:string[32]='Copyright by SKOGOOT SOFTWARE 90';
14 :
15 :  const xres      = 639;     { - pozioma rozdzielczosc komputera  }
16 :        yres      = 399;     { - pionowa rozdzielczosc komputera  }
17 :        accel     = 0.04;    { - przyspieszenie z klawiatury      }
18 :        keyup     = #240;    { - kody klawiszy kursorow           }
19 :        keydown   = #241;    {   charakterystyczne dla            }
20 :        keyright  = #243;    {   AMSTRADA                         }
21 :        keyleft   = #242;
22 :        air       = 0.999;   { - liniowy opor powietrza           }
23 :        gravit    = -0.01;   { - przyspieszenie grawitacyjne      }
24 :
25 :  var   xx,yy,               { - polozenie aktualne               }
26 :        sx,sy    : real;     { - poprzednie polozenie             }
27 :        px,py    : real;     { - predkosc                         }
28 :        ch       : char;
29 :
30 :  procedure SpeedKey(del,rep:byte);
31 :  { ********************************
32 :    procedura pomocnicza ustawiajaca
33 :    czestosc czytania klawiatury
34 :    ********************************
35 :  }
36 :  begin
37 :    inline($3A/del/$67/$3A/rep/$6F/$CD/$5A/$FC/$3F/$BB);
38 :  end; { of SpeedKey }
39 :
40 :  procedure Clg;
41 :  { ********************************
42 :    czyszczenie ekranu graficznego
43 :    ********************************
44 :  }
45 :  begin
46 :     inline($CD/$5A/$FC/$DB/$BB);
47 :  end;
48 :
49 :  function Inkey:char;
50 :  { ********************************
51 :    czytanie klawiatury, proc. podaje
52 :    kod nacisnietego klawisza
53 :    ********************************
54 :  }
55 :   var b:byte;
56 :       press:boolean;
57 :  begin
58 :   inline($CD/$5A/$FC/$09/$BB/$32/b/$D0/$3E/$FF/$32/press);
59 :   if press then Inkey:=chr(b)
```

```
 60 :           else Inkey:=#0;
 61 :  end; { of Inkey }
 62 :
 63 :  procedure Plot(x,y:integer;pc:byte);
 64 :  { *********************************
 65 :    zapala wskazany punkt
 66 :    *********************************
 67 :  }
 68 :  begin
 69 :    inline($3A/pc/$CD/$5A/$FC/$DE/$BB/$2A/x/$EB/$2A/y/$CD/$5A/
 70 :           $FC/$EA/$BB);
 71 :  end; { of Plot }
 72 :
 73 :  procedure Mode(md:byte);
 74 :  { *********************************
 75 :    ustawia tryb graficzny ekranu,
 76 :    typowa dla AMSTRADA CPC
 77 :    *********************************
 78 :  }
 79 :  begin
 80 :   inline($3A/md/$CD/$5A/$FC/$0E/$BC);
 81 :  end; { of Mode }
 82 :
 83 :  procedure Init;
 84 :  { *********************************
 85 :    inicjuje zmienne, ustawia ekran
 86 :    i klawiature
 87 :    *********************************
 88 :  }
 89 :  begin
 90 :   write(#$1B,'0',#$1B,'y');
 91 :   Mode(1); Clg;
 92 :   SpeedKey(1,1);
 93 :   xx:=xres/2;
 94 :   yy:=yres/2;
 95 :   px:=2;
 96 :   py:=1;
 97 :   sx:=xx;
 98 :   sy:=yy;
 99 :  end; { of Init }
100 :
101 :  procedure Bye;
102 :  { *********************************
103 :    pozwala powrocic do normalnego
104 :    trybu pracy w CP/M
105 :    *********************************
106 :  }
107 :  begin
108 :   Mode(2);
109 :   write(#$1B,'1',#$1B,'x');
110 :   SpeedKey(20,2);
111 :  end; { of Bye }
112 :
113 :  begin                      { GLOWNY PROGRAM ****************** }
114 :   Init;
115 :   read(KBD,ch);
116 :   while not (ch='z') do     { nacisniecie 'z' zatrzymuje program }
117 :        begin
118 :        ch:=Inkey;
119 :
120 :        xx:=xx+px;            { nowe polozenie obiektu i odbicia   }
121 :        yy:=yy+py;            { od scian. (ponizej)                }
122 :
123 :        if ((xx+px)>xres) or((xx+px)<0) then px:=-px;
124 :        if ((yy+py)>yres) or((yy+py)<0) then py:=-py;
125 :
126 :        px:=px*air;           { dzialanie oporu powietrza          }
127 :        py:=py*air;
128 :
129 :        py:=py+gravit;        { dzialanie grawitacji               }
130 :
131 :        case ch of
132 :             keyup    :   py:=py+accel;
133 :             keydown :    py:=py-accel;
134 :             keyright:    px:=px+accel;
135 :             keyleft :    px:=px-accel;
136 :             end;
137 :
138 :        if not ((trunc((sx+0.5)/2)=trunc((xx+0.5)/2)) and
139 :               (trunc((sy+0.5)/2)=trunc((yy+0.5)/2))) then
140 :          begin
141 :          Plot(trunc(xx),trunc(yy),1);
142 :          Plot(trunc(sx),trunc(sy),0);
143 :          sx:=xx; sy:=yy;
144 :          end;
145 :        end; { of WHILE not ch='z'... }
146 :   Bye;
147 :  end.                       { KONIEC PROGRAMU ***************** }
```

Kiedy będzie Wielkanoc?

W związku z nadchodzącymi Świętami Wielkanocnymi i lekturą pasjonującej książki dr Michała Szurka pod tytułem „Opowieści Matematyczne", nasunął mi się pomysł napisania programu obliczającego daty Wielkanocy. Program wykorzystuje algorytm Gaussa opisany w tej książce.

Program został napisany na IBM PC, ale biorąc pod uwagę jego prostotę, nie będzie żadnych problemów z uruchomieniem na dowolnym komputerze.

Andrzej Gregorczyk

```
Listing programu 'WIELKANOC'

100 '
110 ' Wielkanoc!
120 ' Autor: Andrzej Gregorczyk
130 ' (c)1990 MK "Bajtek"
140 '
150 CLEAR : DEFINT a-z
160 CLS
170 PRINT " PROGRAM OBLICZA DATĘ WIELKANOCY DLA"
180 PRINT TAB(14);"LAT 1583-3000" : PRINT
190 INPUT "Podaj rok poczatkowy";rp
200 IF rp<1583 OR rp>3000 THEN GOTO 190
210 INPUT "Podaj rok końcowy";rk
220 IF rk<rp OR rk>3000 THEN GOTO 210
230 INPUT "Wydruk (t/n)";dr$
240 PRINT
250 FOR rok=rp TO rk
260 d1=rok : d2=19 : GOSUB 510
270 a=res : d2=100 : GOSUB 510
280 b=il : c=res : d1=b : d2=4
290 GOSUB 510
300 d=il : e=res : d1=b+8 : d2=25
310 GOSUB 510
320 f=il : d1=b-f+1 : d2=3
330 GOSUB 510
340 g=il : d1=19*a+b-d-g+15 : d2=30
350 GOSUB 510
360 h=res : d1=c : d2=4
370 GOSUB 510
380 j=il : k=res : d1=32+2*e+2*j-h-k
390 d2=7 : GOSUB 510
400 s=res : d1=a+11*h+22*s : d2=451
410 GOSUB 510
420 u=il : d1=h+s-7*u+114 : d2=31
430 GOSUB 510
440 PRINT
450 IF il=3 THEN m$="marca" ELSE m$="kwietnia"
460 PRINT "Rok";rok;" Wielkanoc:  ";res+1;m$
470 IF dr$<>"T" AND dr$<>"t" THEN 490
480 LPRINT "Rok";rok;" Wielkanoc:  ";res+1;m$
490 NEXT rok
500 END
510 il=INT(d1/d2) : res=d1-d2*il
520 RETURN
```

```
Rok 1990  Wielkanoc:  15 kwietnia
Rok 1991  Wielkanoc:  31 marca
Rok 1992  Wielkanoc:  19 kwietnia
Rok 1993  Wielkanoc:  11 kwietnia
Rok 1994  Wielkanoc:  3 kwietnia
Rok 1995  Wielkanoc:  16 kwietnia
Rok 1996  Wielkanoc:  7 kwietnia
Rok 1997  Wielkanoc:  30 marca
Rok 1998  Wielkanoc:  12 kwietnia
Rok 1999  Wielkanoc:  4 kwietnia
Rok 2000  Wielkanoc:  23 kwietnia
Rok 2001  Wielkanoc:  15 kwietnia
Rok 2002  Wielkanoc:  31 marca
Rok 2003  Wielkanoc:  20 kwietnia
Rok 2004  Wielkanoc:  11 kwietnia
Rok 2005  Wielkanoc:  27 marca
Rok 2006  Wielkanoc:  16 kwietnia
Rok 2007  Wielkanoc:  8 kwietnia
Rok 2008  Wielkanoc:  23 marca
Rok 2009  Wielkanoc:  12 kwietnia
Rok 2010  Wielkanoc:  4 kwietnia
Rok 2011  Wielkanoc:  24 kwietnia
Rok 2012  Wielkanoc:  8 kwietnia
Rok 2013  Wielkanoc:  31 marca
Rok 2014  Wielkanoc:  20 kwietnia
Rok 2015  Wielkanoc:  5 kwietnia
Rok 2016  Wielkanoc:  27 marca
Rok 2017  Wielkanoc:  16 kwietnia
Rok 2018  Wielkanoc:  1 kwietnia
Rok 2019  Wielkanoc:  21 kwietnia
Rok 2020  Wielkanoc:  12 kwietnia
Rok 2021  Wielkanoc:  4 kwietnia
Rok 2022  Wielkanoc:  17 kwietnia
Rok 2023  Wielkanoc:  9 kwietnia
Rok 2024  Wielkanoc:  31 marca
Rok 2025  Wielkanoc:  20 kwietnia
Rok 2026  Wielkanoc:  5 kwietnia
Rok 2027  Wielkanoc:  28 marca
Rok 2028  Wielkanoc:  16 kwietnia
Rok 2029  Wielkanoc:  1 kwietnia
Rok 2030  Wielkanoc:  21 kwietnia
```

JAK ZŁOŻYĆ (TANI) KOMPUTER IBM?

Poniższe porady adresowane są do tych Czytelników, którzy chcieliby mieć komputer klasy IBM, a nie chcą kupować gotowego lub nie mogą sobie na niego pozwolić.

Samodzielna budowa opłaca się finansowo i nie wymaga specjalnej wiedzy elektronicznej, potrzebne są jednak pewne minimalne zdolności manualne i techniczne. Jeśli po przeczytaniu poniższego opisu stwierdzisz, że niewiele rozumiesz, lepiej nie podejmuj próby samodzielnego składania komputera, by uniknąć niepotrzebnych rozczarowań i wydatków.

Podstawą sukcesu jest tanie kupno potrzebnych elementów. Nie jest prawdą, że najtaniej jest na giełdach. Wiele firm zajmujących się sprzedażą komputerów bardzo często sprzedaje również podzespoły, których ceny są porównywalne z giełdowymi, a zakupiony sprzęt posiada gwarancję. Kupowanie na giełdach wymaga pewnego wyczucia. Należy wybierać rzeczy co do których nie ma się wątpliwości, że są nowe i sprawne. Zanim zaczniesz kupować zorientuj się więc w cenach (te podane przeze mnie były aktualne w grudniu 1990 r.).

Załóżmy, że chcesz złożyć komputer klasy XT w minimalnej konfiguracji (można będzie zawsze ją rozszerzyć). Minimum, niezbędnym do uruchomienia jakiegokolwiek programu, jest 256 KB pamięci RAM i jeden napęd dysków elastycznych.

Niezbędne są następujące podzespoły (*patrz też ELEMENTarz PC w poprzednim numerze „Bajtka" —MB*):

1. Płyta główna. Istnieją zasadniczo dwa typy płyt głównych. Pierwsza z nich posiada wymiary kartki A4 i jest zbudowana ze standardowych układów scalonych. Druga, dzięki zastosowaniu mniejszej ilości bardziej specjalizowanych układów, jest nieco mniejsza. Jeśli zależy nam na minimalnych kosztach, to warto kupić płytę bez zamontowanych kości pamięci (tzw. 0 RAM). Kostki pamięci z łatwością kupimy tanio na giełdzie lub z ogłoszenia. Należy jednak zwrócić uwagę na to, że małe płyty główne często są wyposażone w pamięci o organizacji 4∗256 KB. Cena tych kości jest ponad czterokrotnie wyższa od kostek o organizacji 1∗256 KB (typowych dla dużych płyt), w efekcie kupno małej płyty jest nieco droższym rozwiązaniem. Potrzebny typ pamięci z łatwością można poznać licząc ilość podstawek przeznaczonych pod kości — płyty z pamięciami 1∗256 KB mają 24 lub 36 podstawek, a z pamięciami 4∗256 KB — 12 podstawek.

Ważnym parametrem, opisującym kości pamięci, jest czas dostępu. Do komputera z TURBO równym 10 MHz w zupełności wystarczają kości z czasem dostępu równym 120 ns (czasem nawet 150 ns).

Jeżeli nie chcesz żałować zakupu, koniecznie sprawdź częstotliwość zegara dla pracy TURBO. Częstotliwość można odczytać z rezonatora kwarcowego, pamiętając o tym, że trzeba ją podzielić przez 3 lub przez 2. Wartość podzielnika można ustalić z wartości częstotliwości dla pracy normalnej. Przykładowo, jeśli na płycie są kwarce 14.31 MHz i 30 MHz, to podzielnik wynosi 3 (bo 14.31/3=4.77) i TURBO wynosi 10 MHz.

Ponadto należy koniecznie policzyć gniazda rozszerzające. Nie warto kupować płyt o ilości gniazd mniejszej od 5. Ostatnią bardzo ważną rzeczą jest opis płyty. Często nawet w poważnych firmach podzespoły są sprzedawane bez żadnej dokumentacji. Należy stanowczo dopominać się opisu, w przeciwnym przypadku można spędzić kilka nocy na ustalaniu miejsc podłączenia głośnika, przycisków TURBO, RESET itd. Orientacyjna cena płyty bez pamięci — 500 tys. zł, cena 640 KB pamięci około 300 tys.

2. Obudowa. Obudowy spotykane na rynku można podzielić na cztery grupy: klasyczne — szerokie i długie, *compact* — wąskie i długie, *baby* — wąskie i krótkie oraz *tower* — pionowe. Najbardziej godnymi polecenia są obudowy *compact*. W obudowie tego typu zmieści się płyta główna mała lub duża, ponadto zawiera ona miejsce na dwie jednostki dysków elastycznych 5.25", jedną jednostkę 3.5" i twardy dysk.

Przed kupieniem obudowy sprawdź koniecznie, czy pasuje do niej kupiona płyta główna (rozstaw otworów). Obudowy zwykle są sprzedawane z zamontowanym zasilaczem (optymalna moc to mniej więcej 150 W, wyższa niepotrzebna). Odradzam kupowanie samej obudowy, gdyż zasilacz trudno kupić jako samodzielny podzespół, w dodatku może on nie pasować gabarytami. Cena obudowy z zamontowanym zasilaczem jest zwykle mniejsza od kosztu podzespołów kupowanych osobno. Orientacyjna cena obudowy *compact* z zasilaczem 200 W — 900 tys. zł.

3. Karta graficzna. Jeśli bardzo zależy Ci na obniżeniu kosztów, możesz zakupić kartę CGA i podłączyć ją do taniego monitora Neptun lub Vela. Najlepszym rozwiązaniem będzie jednak kupno karty Hercules. Istnieje wiele typów tych kart. Nowe produkty wykonane są zwykle jako karty krótkie i zawierają specjalizowane układy scalone. Lepiej kupić taką kartę, która posiada ROM z matrycą znaków umieszczony w podstawce. Pozwoli to na jego łatwą wymianę i instalację polskich znaków. Karty, które na płytce zawierają tylko jeden specjalizowany układ scalony i nic więcej, nie mają tej możliwości. Orientacyjne ceny: Hercules — 200 tys., CGA — 100 tys.

4. Karta Multi I/O. Powinna zawierać sterownik dwóch napędów dysków elastycznych, zegar z podtrzymaniem bateryjnym i jeden lub dwa interfejsy RS232. Spotykane są też karty zawierające oprócz tego jeden uniwersalny port równoległy i port joysticka oraz takie, które zamiast standardowego sterownika dysku 2∗360 KB mają sterownik 1∗1.2 MB i 1∗360 KB. Kupno takiej karty szczególnie można polecić tym, którzy nie planują w przyszłości zakupu twardego dysku. Do karty powinien być dołączony kabel z trzema wtykami — dwa na końcach i jeden w środku, ze zwiniętą w jednym miejscu taśmą. Cena zwykłej karty około 300 tys., ze sterownikiem napędu 1.2 MB około 400 tys.

5. Napęd dysków. Na tym nie warto oszczędzać — zakup napędu renomowanych firm jak TEAC, Panasonic, BASF odpłaci się cichą, bezawaryjną pracą. Do uruchomienia komputera wystarczy jeden napęd. Orientacyjna cena — 650 tys.

6. Klawiatura. Wybór typu pozostawiam Czytelnikowi; zależy on głównie od indywidualnych upodobań. Warto jednak przypomnieć, że jakość klawiatury ma bezpośredni wpływ na komfort pracy z komputerem. Cena jest zależna od typu i waha się od 200 do 400 tys. zł.

7. Monitor. Do karty CGA wystarczy nasz Neptun 156. Jakość obrazu nie dorównuje monitorom profesjonalnym, ale na początek jest zadawalająca. W przypadku karty Hercules kupno lepszego monitora jest konieczne. Trudno polecić konkretny typ i producenta, tym bardziej że parametry i ceny różnych egzemplarzy są zbliżone. Orientacyjna cena 900–1300 tys.

Przybliżona cena całego komputera z kartą Hercules i monitorem wyniesie zatem około 4200 tys. złotych. Warto jednak zauważyć, że ceny podzespołów stale maleją, aktualnie cena zestawu może być więc jeszcze niższa.

Jeśli już skompletujesz potrzebne podzespoły, to w przeciągu jednego popołudnia można poskładać je w jedną całość. Nie są w tym celu potrzebne żadne specjalne narzędzia — w zupełności wystarczy wkrętak, pęseta i szczypce. Niestety, potrzebne będą dwa rodzaje śrub, gdyż część podzespołów wymaga do umocowania śrub z gwintem metrycznym, a część z gwintem calowym.

Zaczynamy od włożenia w podstawki kości pamięci. Kości muszą być tak włożone, by wcięcia na krótszych bokach podstawek i kości były z tej samej strony. Następna rzecz — to przymocowanie płyty głównej do obudowy i podłączenie jej do zasilacza. Wtyczki zasilania można włożyć tylko prawidłowo — dotyczy to również napędów dyskietek, które trzeba mocno przykręcić w odpowiednim miejscu obudowy. Następnie trzeba w gniazda włożyć karty rozszerzające, tak je rozmieszczając, by wieloprzewodowe taśmy do napędów nie utrudniały w przyszłości dostępu do żadnej karty. Najprościej można to osiągnąć wkładając kartę Multi I/O w gniazdo najbliższe stacji, a kartę graficzną gdzieś dalej.

Drobnego wyjaśnienia wymaga podłączenie napędu dysków. Taśma wieloprzewodowa ma oznaczony kolorem czerwonym przewód nr 1, również szpilka numer 1 na karcie powinna być w jakiś sposób oznaczona — np. napisanym numerem lub kwadratową podstawą. Podobne zasady dotyczą drugiego końca kabla — czyli przyłączanego do stacji. Wtyk na samym końcu taśmy należy podłączyć do napędu A, a wtyk w połowie długości taśmy — do napędu B. Jeżeli dysponujemy tylko jedną stacją, po prostu rezygnujemy z napędu B, bez żadnych dodatkowych operacji. Same napędy powinny mieć ustawioną drugą zworkę — w zależności od przyjętego oznaczenia będzie ona miała numer 1 (jeśli zworki numerowane są od zera), 2 (jeśli są numerowane od jednego) lub B (jeśli przyjęte jest oznaczenie literowe). Ostatni napęd na kablu (czyli zawsze napęd A) powinien mieć założoną drabinkę rezystorową — zwykle i drabinka, i zworka są ustawiane fabrycznie w taki sposób, że ingerencji (OSTROŻNEGO usunięcia drabinki) wymaga tylko napęd B.

Komputer złożony z dobrych podzespołów zaczyna działać od pierwszego włączenia zasilania, czego życzę wszystkim moim naśladowcom.

Robert Magdziak

FORMATOWANIE DYSKIETEK



Na Spectrum sprawa była prosta — wkładało się taśmę do magnetofonu, naciskało REC-PLAY i po chwili program był nagrany — co najwyżej można go było jeszcze zweryfikować. Czasy i komputery się zmieniają — dzisiaj siedzę przed PC- etem, który zapisuje informacje na dyskietkach. Wydanie komendy SAVE przestało wystarczać — najpierw trzeba trochę poczarować, a przede wszystkim — sformatować dyskietkę.
Po co to formatowanie? Co to, nie widać że dyskietka ma formę okrągłą, z dziurą w środku i kanciastą kopertą dookoła?

Opcje programu **format.com** (MS-DOS 3.30):

/1	-	formatowanie tylko jednej strony dyskietki;
/4	-	formatowanie dyskietki 5.25" na 360 KB w stacji 1.2 MB;
/8	-	formatowanie 8 (zamiast standardowych 9) sektorów na ścieżce;
/b	-	zostawia na dyskietce miejsce na system operacyjny;
/s	-	kopiuje na dyskietkę system operacyjny;
/t:xx	-	podawanie liczby formatowanych ścieżek;
/n:xx	-	podawanie liczby formatowanych sektorów;
/v	-	po sformatowaniu dyskietki można nadać jej nazwę.

Opcje, z których można korzystać, zależą od stacji, w której będzie formatowana dyskietka. Błąd w doborze parametrów sygnalizowany jest komunikatem *Parameters not supported.*

Podstawowe sposoby formatowania dyskietek:

format a:	- formatowanie dyskietki w stacji **a:**, dyskietka będzie sformatowana zgodnie z typem stacji.
format a:/4	- formatowanie dyskietki 360 KB w stacji 1.2 MB, dyskietka może się nie dać odczytać w stacji niskiej gęstości.
format a:/n:9	- formatowanie dyskietki 720 KB w stacji 1.44 MB.

W stacji 360 KB nie da się sformatować dyskietki sformatowanej wcześniej na 1.2 MB. Trzeba ją najpierw sformatować na 360 KB przez **format a: /4** w stacji 1.2 MB.

Uwaga: **format.com** w DOS-ach wcześniejszych niż 3.00 w razie nie podania stacji, w której ma być formatowana dyskietka, formatuje dyskietkę w aktualnym napędzie - może to być twardy dysk!

Pierwsze formatowanie twardego dysku jest trójetapowe. Najpierw wykonuje się tzw. twarde formatowanie (za pomocą programu zawartego w BIOS-ie kontrolera), następnie przygotowuje się informacje o partycjach za pomocą programu **fdisk.com** (nie do ominięcia, nawet jeśli na dysku ma być tylko jedna partycja), i dopiero na końcu można użyć programu **format.com**. W razie konieczności ponownego sformatowania twardego dysku (np. po zniszczeniu FAT-u przez wirusa) pierwszy etap (czasem i drugi) można pominąć.

Widać — nie widać, sformatować trzeba. Zanim jednak zajmiemy się samym formatowaniem, przyjrzyjmy się różnicom, jakie istnieją między taśmą a dyskietką. Najważniejsza dotyczy metody dostępu do informacji — na taśmie możemy nasze dane przeglądać tylko w takiej kolejności, w jakiej zostały na niej zapisane, toteż jeśli chcemy sprawdzić jej zawartość, musimy przeczytać wszystko, co się na niej znajduje po kolei. Mówimy, że dostęp do danych zapisanych na taśmie jest **sekwencyjny**. Z dyskietkami jest inaczej — ponieważ informacje są zapisane w różnych miejscach powierzchni krążka, możemy ustawiając głowicę nad odpowiednio wybranym fragmentem dyskietki odczytać dowolną potrzebną informację. Jest to dostęp **swobodny** — i nikomu, kto przewijał całą taśmę w tę i z powrotem, żeby znaleźć JETPAC-a, nie trzeba tłumaczyć, jaka jest różnica szybkości między obydwoma rozwiązaniami.

Żeby jednak dyskietka mogła spełniać swoje zadania, zapisana na niej informacja musi być odpowiednio zorganizowana. Najczęściej stosowanym sposobem jest podzielenie powierzchni dyskietki na niewielkie obszary, na których zapisuje się dane. W trakcie pracy dyskietka kręci się, a głowica odczytująca informacje ślizga się po jej powierzchni. Cały obszar, który może zostać odczytany przez głowicę podczas jednego obrotu dyskietki, nazywa się **ścieżką**. Ponieważ jednak na powierzchni ścieżki można zapisać stosunkowo dużą ilość informacji, wygodnie jest ją podzielić na pewne niewielkie fragmenty, nazywane **sektorami**. Sektory są podstawowymi elementami każdego zbioru zapisanego na dysku, niezależnie od komputera. Żeby umożliwić korzystanie z nich, trzeba je jakoś ponumerować. Najtrudniej zaznaczyć pierwszy sektor, gdyż pozostałe mogą być ponumerowane kolejno. Do tego celu służy niewielki otwór w każdej dyskietce, rozpoznawany przez stację za pomocą fotodiody.

Fizyczna konstrukcja stacji zawierających więcej niż jedną głowicę — co pozwala na dostęp do obydwu stron dyskietki i podwojenie ilości danych, do których mamy dostęp — powoduje, że istnieją jeszcze dwie struktury, jakie można wskazać na każdej dyskietce (i twardym dysku). Pierwsza to **cylinder**, czyli wszystkie ścieżki, na wszystkich stronach dysku, dostępne przy danym ustawieniu głowic, druga to **klaster** (z ang. *cluster*) — wszystkie sektory w danym cylindrze, mające ten sam numer.

Oczywiście samo podzielenie powierzchni dyskietki na pewne obszary nie pozwala jeszcze na jej wykorzystanie — w tym celu trzeba zaprojektować jakiś sposób zapamiętywania informacji o tym, co się w którym sektorze znajduje i — co również bardzo ważne — czy sektor w ogóle nadaje się do użytku. Może się bowiem tak zdarzyć, że z powodu niewielkiego uszkodzenia powierzchni, w pewnym miejscu dyskietki nie można tak zapisać informacji, by dało się ją później odczytać. Pogiętą taśmę na ogół trzeba wyrzucić, w przypadku dysku wystarczy gdzieś zapamiętać informację o tym, że sektor jest uszkodzony i nie wolno z niego korzystać. W przypadku MS DOS-u wszystkie wymienione wyżej informacje zawarte są w dwóch strukturach danych — jedna nazywa się **FAT** (*File Allocation Table*), druga to **katalogi**. (Dokładnie całą dyskietkę DOS-u opisał w klanie Amstrada w

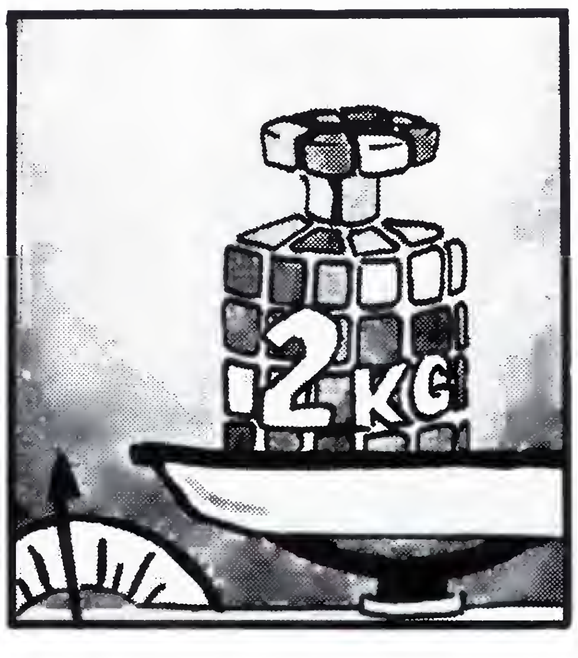
numerze III '90 Jonasz Mayer i do jego artykułu odsyłam zainteresowanych szczegółami).

Teraz doszliśmy wreszcie do formatowania dyskietek. Zanim system operacyjny może dyskietkę wykorzystać, musi mieć na niej przygotowane FAT i miejsce na główny katalog. Ich przygotowanie wymaga przetestowania całej powierzchni dyskietki i zapisania w określonych miejscach niezbędnych dla systemu informacji. Te właśnie operacje wykonuje program formatujący **format.com.**

Najczęściej spotykanymi w PC-etach rozmiarami dyskietek są 5.25" i 3.5" (oraz niewielkie ilości stacji 3" w laptopach). Dyskietkę 5.25" można sformatować tak, by dawało się na niej zapisać 360 KB lub 1.2 MB informacji (można i inaczej, ale to są dwa najpopularniejsze standardy). Od czego to zależy? Pojemność dyskietki (co nietrudno sprawdzić) jest równa iloczynowi ilości stron (ta jest właściwie stała i równa 2), ilości ścieżek na stronie ilości sektorów w obrębie ścieżki i ilości bajtów w sektorze (też zwykle stała i równa 512). Podstawowy format, 360 KB, to efekt połączenia 40 ścieżek na stronie z 9 sektorami w jednej ścieżce. Tyle można było bezpiecznie wycisnąć ze stacji i dyskietek na początku lat osiemdziesiątych. Dość szybko jednak udało się poprawić jakość dyskietek na tyle, że na jednej stronie dyskietki dawało się wcisnąć 80 węższych ścieżek zawierających po 15 sektorów, co dawało pojemność dyskietki równą 1.2 MB. Jednak węższa ścieżka oznacza inną konstrukcję głowicy, czyli zupełnie nową stację. Stare stacje, z szerokimi głowicami, nie są w stanie przeczytać dyskietek 1.2 MB. W drugą stronę problemu nie ma — w stacji o dużej gęstości dyskietki 360 KB można przeczytać bez trudu. Co więcej, ustawiając w odpowiednim miejscu wąską głowicę, można w stacji 1.2 MB zapisać informacje na dyskietce 360 KB w taki sposób, że dadzą się odczytać w starej stacji. Tak przynajmniej mówi teoria. W praktyce na ogół rzeczywiście tak się dzieje, ale od czasu do czasu zdarzają się kłopoty z odczytywaniem dyskietek 360 KB zapisywanych w stacjach 1.2 MB, toteż lepiej tego w miarę możliwości unikać.

Nieco inaczej ma się sprawa z dyskietkami 3.5". Tu również istnieją dwa podstawowe formaty — 720 KB i 1.44 MB, ale różnica między nimi nie polega na ilości ścieżek, która wynosi 80 w obu formatach, ale na ilości sektorów. Na dyskietkach 720 KB sektorów jest 9 na każdej ścieżce, na dyskietkach 1.44 MB — 18. Większa ilość sektorów oznacza większe wymagania w stosunku do szybkości komputera, który musi wczytać komplet informacji w czasie jednego obrotu dyskietki, toteż stacji 1.44 MB nie da się bezpośrednio zainstalować w standardowym XT (to samo dotyczy stacji 1.2 MB). Ponieważ jednak niezależnie od stosowanej stacji (720/1.44) parametry zapisu (szerokość ścieżki, długość sektora) na dyskietce sformatowanej na 720 KB są identyczne, dyskietki 3.5" pozwalają na bezpieczne przenoszenie danych między komputerami, niezależnie od rodzaju stacji, jakimi dysponujemy.

Marcin Borkowski



—KOSTKA—

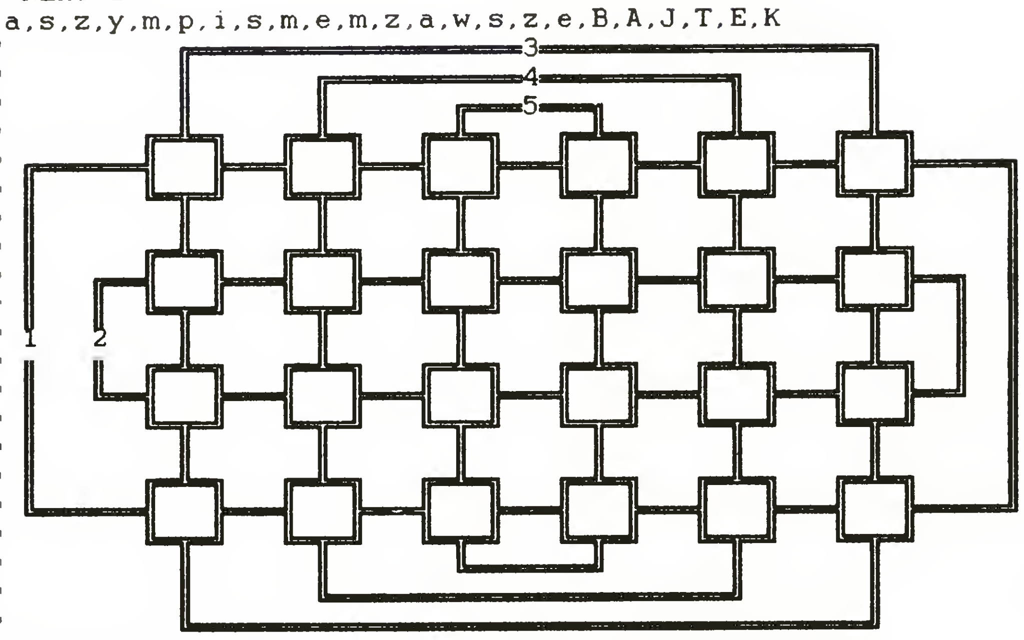
CIĘŻARKA

Kilka lat temu wszyscy się bawili kostką Rubika, a mój brat cioteczny nawet wygrał konkurs na szybkość jej układania w Teleranku. Mało sobie palców nie połamał na tej szybkości. Taki jesteś mądrala? To dzisiaj mam dla Ciebie kostkę Ciężarka, zobaczymy, jak sobie dasz radę.

Zadanie jest bardzo proste — za pomocą pięciu możliwych przesunięć należy ułożyć układankę — czyli napis opiewający cudowność „Bajtka" i deklarujący wierność temu tytułowi. Nie potrzeba żadnych Pascali ani C, ani nic — tylko stary dobry BASIC w dowolnym PC-etowym wydaniu. Podanie numeru spowoduje wykonanie ruchu, czyli przesunięcie wszystkich liter znajdujących się na bokach prostokąta o podanym numerze o jedną pozycję, zgodnie z kierunkiem wskazówek zegara. Żeby nikogo nie denerwować, ruchy nie są liczone, nie ma również limitu czasu. Po dwóch godzinach wskazane jest naciśnięcie trzech klawiszy Ctrl-Alt-Del naraz, w przeciwnym przypadku może nastąpić przegrzanie się (gracza, nie komputera). Powodzenia, braciszku...

Marek Ciężarek

```
1000 GOSUB 1110
1010 FOR I=1 TO 4 : FOR J=1 TO 6
1020   LOCATE 5+4*(I-1),3+8*J : PRINT A$(I,J)
1030 NEXT J : NEXT I
1040 LOCATE 22,10 : INPUT "Numer ruchu";WYBOR$
1050 LOCATE 22,22 : PRINT ,
1060 IF WYBOR$>"5" OR WYBOR$<"0" THEN GOTO 1040
1070 RUCH=VAL(WYBOR$)
1080 ON RUCH+1 GOSUB 1450,1490,1490,1580,1580,1580
1090 GOTO 1010
1100 STOP
1110 REM
1120 REM przygotowanie danych, wydrukowanie ramek
1130 REM
1140 CLS : DIM A$(4,6)
1150 FOR I=1 TO 4 : FOR J=1 TO 6
1160   READ A$(I,J)
1170 NEXT J: NEXT I
1180 DATA n,a,s,z,y,m,p,i,s,m,e,m,z,a,w,s,z,e,B,A,J,T,E,K
1190 PRINT "                                                  "
1200 PRINT "                                                  "
1210 PRINT "                                                  "
1220 PRINT "                                                  "
1230 PRINT "                                                  "
1240 PRINT "                                                  "
1250 PRINT "                                                  "
1260 PRINT "                                                  "
1270 PRINT "                                                  "
1280 PRINT "                                                  "
1290 PRINT " 1  2                                             "
1300 PRINT "                                                  "
1310 PRINT "                                                  "
1320 PRINT "                                                  "
1330 PRINT "                                                  "
1340 PRINT "                                                  "
1350 PRINT "                                                  "
1360 PRINT "                                                  "
1370 PRINT "                                                  "
1380 PRINT "                                                  "
1390 PRINT "                                                  "
1400 FOR I=1 TO 5+INT(10*RND) : REM pomieszanie w tablicy
1410   RUCH=1+INT(4*RND)
1420   ON RUCH GOSUB 1490,1490,1580,1580,1580
1430 NEXT I
1440 RETURN
1450 REM
1460 REM zakończenie gry
1470 REM
1480 CLS : PRINT "Do widzenia!" : SYSTEM
1490 REM
1500 REM przewijanie poziome liter
1510 REM
1520 B$=A$(RUCH,6)
1530 FOR I=6 TO 2 STEP -1 : A$(RUCH,I) = A$(RUCH,I-1) : NEXT I
1540 A$(RUCH,1)=A$(5-RUCH,1)
1550 FOR I=1 TO 5 : A$(5-RUCH,I)=A$(5-RUCH,I+1) : NEXT I
1560 A$(5-RUCH,6)=B$
1570 RETURN
1580 REM
1590 REM przewijanie pionowe liter
1600 REM
1610 RUCH=RUCH-2 : B$=A$(1,RUCH)
1620 FOR I=1 TO 3 : A$(I,RUCH)=A$(I+1,RUCH) : NEXT I
1630 A$(4,RUCH)=A$(4,7-RUCH)
1640 FOR I=4 TO 2 STEP -1 : A$(I,7-RUCH)=A$(I-1,7-RUCH) : NEXT I
1650 A$(1,7-RUCH)=B$
1660 RETURN
```



RAMY, RAMKI, RAMECZKI...

Marek opisując swoją kostkę Ciężarka zapomniał o tym, że nie wszyscy wiedzą, jak się rysuje ramki.

218	194	191	─	196
195	197	180	│	179
192	193	217	═	205
			║	186
201	203	187		
204	206	185	░	176
200	202	188	▒	177
			▓	178
214	210	183	█	219
199	215	182		
211	208	189	▀	223
			▄	220
213	209	184		
198	216	181	▌	221
212	207	190	▐	222

W standardowym zestawie znaków IBM-a istnieje cały komplet służący do tworzenia nawet skomplikowanych ramek i tabelek. Samo ich istnienie to jednak za mało — trzeba jeszcze umieć z nich skorzystać. Wymaga to po pierwsze znajomości odpowiednich kodów ASCII, po drugie — umiejętności ich wprowadzania. Kody ASCII znajdziecie tuż obok — nie próbujcie dopatrzeć się w nich żadnej regularności, lepiej mieć je gdzieś pod ręką, niż starać się zapamiętać. Wprowadza się je (z klawiatury) następująco — należy nacisnąć klawisz **alt** i nie puszczając go wystukać odpowiedni kod ASCII na klawiaturze numerycznej (nie ma znaczenia, czy jest ona w trybie NumLock, czy nie). Po puszczeniu klawisza **alt** na ekranie pojawi się potrzebny znaczek (chyba że edytor, z którego korzystacie, ma na ten temat swoje własne zdanie — np. w Turbo Pascalu trzeba operację przeprowadzić możliwie szybko, bo nic z tego nie wyjdzie). W ten sposób można wprowadzić z klawiatury każdy znak o dowolnym kodzie — oprócz zera.

Marcin Borkowski

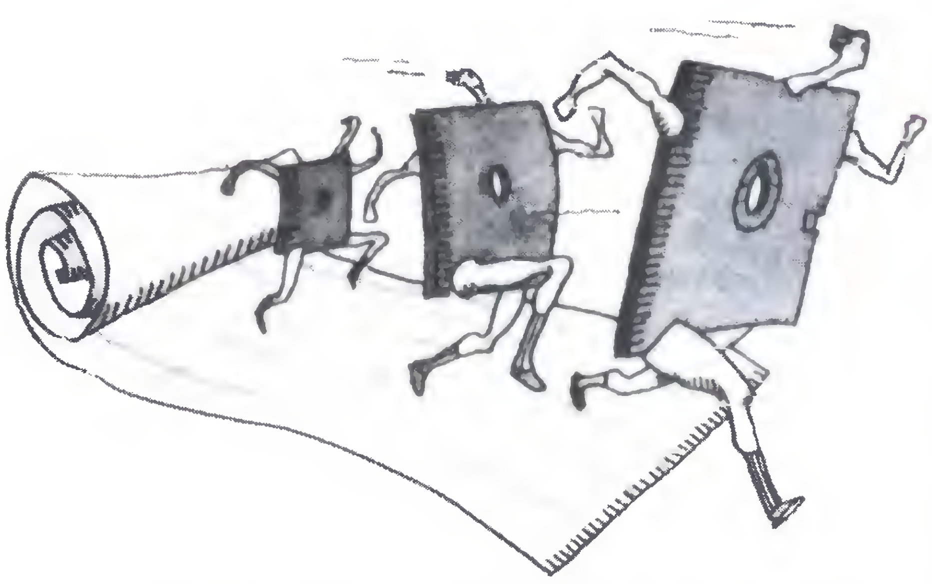
W GŁĄB DOS-u

W poprzednim numerze opisałem z grubsza jak jest zbudowany PC-et, co w nim musi i może siedzieć w środku. Nietrudno się domyślić, że Najbardziej Udana Konstrukcja wymaga oprogramowania, bez którego ani rusz.

Ponieważ jesteśmy w klanie IBM-a, przyjrzyjmy się, jak zostało zorganizowane oprogramowanie tej maszyny. Wiemy już z poprzedniego numeru co nieco na temat jej budowy — miesiąc temu pisałem (*ELEMENTarz PC*) między innymi o tym, że na płycie głównej znajduje się ROM

BIOS,

czyli pamięć z zestawem podprogramów służących do wykonywania najbardziej podstawowych operacji na urządzeniach zewnętrznych. Te podstawowe operacje w większości służą do komunikacji z otoczeniem, czyli są to operacje wejścia-wyjścia i stąd skrót BIOS — **Basic Input Output System**. Ta najgłębsza warstwa oprogramowania, oprócz tego, że bez wykonywanych przez nią obowiązków (czytanie znaków z klawiatury, wypisywanie ich na ekranie, czytanie pojedynczych sektorów dyskietki, wysyłanie znaków na drukarkę itd.) nie sposób opanować komputera, ma jeszcze jedno zadanie do wykonania. Jest nim zapewnienie kompatybilności na poziomie oprogramowania.



Konstruktor, pragnący zbudować komputer zgodny z IBM PC, często staje przed wyborem — może albo powtórzyć rozwiązania zastosowane w oryginalnym komputerze, albo usprawnić je, zwiększając szybkość działania swojej konstrukcji, na przykład przez zastosowanie innych elementów — niestety za cenę utraty pełnej kompatybilności sprzętowej. Dla większości programów nie będzie to miało znaczenia, gdyż nie odwołują się one do sprzętu bezpośrednio, lecz korzystają z usług oferowanych przez BIOS (lub jeszcze wyższe piętra oprogramowania). Jeśli więc usługi BIOS-u będą w tym nieco zmienionym komputerze identyczne z oryginalnym IBM-em, większość programów będzie działała prawidłowo. To właśnie będzie kompatybilność na poziomie oprogramowania.

Jak ją zrealizowano? Wykorzystano w tym celu pewną cechę zastosowanego w konstrukcji PC-eta procesora 8088. Na samym dole pamięci komputera znajduje się długa tablica, zawierająca adresy wielu podprogramów usługowych. Można je wywoływać wydając procesorowi odpowiednią komendę i podając mu numer potrzebnego w danej chwili podprogramu. Procesor sam znajdzie w tablicy potrzebny adres i wykona odpowiedni podprogram, a potem wróci do tego miejsca, w którym podprogram był potrzebny. Z pewnych powodów podprogramy te nazywa się **przerwaniami**, a tablicę adresów — **wektorami przerwań**. Operacje wykonywane przez poszczególne przerwania, oraz sposoby przekazywania danych do i z podprogramu są bardzo precyzyjnie ustalone. Umożliwia to ich wykorzystanie, a dodatkowo takie rozwiązanie pozwala każdemu zainteresowanemu napisać własne podprogramy — po ich umieszczeniu w pamięci komputera i zmianie właściwego wektora przerwań, komputer będzie odpowiednie operacje wykonywać tak jak chcemy — na przykład nie dopuści do zapisania czegokolwiek na dysku.

Następnym poziomem oprogramowania, niezależnym już od sprzętu, jest właściwy system operacyjny, czyli — w przypadku PC-eta —

PC DOS

lub MS DOS. Jest to prawie dokładnie ten sam system, różnica polega na tym, że PC DOS jest rozprowadzany przez IBM, a MS DOS przez Mocrosoft i tych producentów komputerów, którzy uzyskali licencję (np. Elwro sprzedaje MS DOS razem z Mazovią).

Co robi DOS? Obsługuje — podobnie do BIOS-u — operacje wejścia-wyjścia, ale na wyższym poziomie. Przykładowo, i BIOS i DOS pozwalają na odczytanie informacji zapisanych na dyskietce, ale BIOS potrafi odczytać je wyłącznie korzystając z fizycznego adresu sektora (składającego się z numerów strony, ścieżki i sektora, patrz tekst o formatowaniu dyskietek w tym klanie), DOS-owi zaś wystarczy nazwa zbioru, w którym informacje się znajdują. Innymi słowy, DOS tłumaczy nasze żądanie odczytania informacji ze zbioru o określonej nazwie na żądanie odczytania konkretnego sektora, po czym zleca to zadanie BIOS-owi. Oczywiście wymaga to wykonywania wielu zadań związanych z administrowaniem systemem i utrzymywaniem porządku.

BIOS znajduje się zwykle w ROM-ie (Read Only Memory — pamięć tylko do odczytu) na płycie głównej. Inaczej jest z DOS-em, który musi zostać załadowany do pamięci operacyjnej z dysku (lub dyskietki) po uruchomieniu komputera, co jest ostatnim zadaniem wykonywanym po włączeniu komputera przez BIOS (po przetestowaniu sprzętu). Pozwala to na łatwą zmianę stosowanego systemu operacyjnego — na tym samym komputerze może to być raz PC DOS 2.11, kiedy indziej MS DOS 4.01, a przy innej okazji nawet UNIX. Istnieją wprawdzie komputery zawierające DOS w ROM-ie, ale rozwiązanie takie stosuje się tylko tam, gdzie chodzi o zminimalizowanie ilości odwołań do dyskietek — np. w komputerach przenośnych.

Mamy więc już w komputerze BIOS i wczytany z dyskietki DOS (składający się, co nie ma dla nas większego znaczenia, z dwóch odrębnych programów). Niestety, ciągle jeszcze nie możemy rozpocząć pracy. Zanim będzie to możliwe, system musi jeszcze wczytać

interpreter poleceń.

Jest to taki program (w przypadku MS i PC DOS-u nazywający się **command.com**), który pozwala na skorzystanie z możliwości DOS-u. Wszystkie usługi systemu operacyjnego są bowiem dostępne tylko za pośrednictwem przerwań, podobnie jak to było w przypadku BIOS-u. Potrzebny jest więc jeszcze jeden program, który wczyta z klawiatury wydany przez nas rozkaz i przetłumaczy go na język przerwań — zwykle jest to właśnie **command.com**, choć istnieje możliwość zastąpienia standardowego interpretera innym, dopasowanym do potrzeb lub gustów użytkownika.

Co potrafi **command.com**? Przede wszystkim uruchomić program, którego nazwę podamy. Ta umiejętność może być wystarczająca — wykonanie każdej operacji potrzebnej w czasie pracy, czy będzie to przejrzenie katalogu, czy sprawdzenie godziny, może być realizowane przez odpowiedni program (tak jest w systemie UNIX). Takie rozwiązanie, choć bardzo proste w realizacji ze względu na niewielkie wymagania stawiane interpreterowi, powoduje dużą ilość operacji dyskowych — przed wykonaniem każdego programu musi on być wczytany z dysku, co znacznie spowalnia pracę systemu. W DOS-ie zdecydowano się więc na nieco inne rozwiązanie. Część poleceń jest wbudowana w interpreter i ich wykonanie polega na wywołaniu znajdującego się w pamięci komputera podprogramu. Te polecenia nazywamy rezydentnymi. Oczywiście nie wszystko można trzymać w pamięci, dlatego też inne polecenia, wykonywane znacznie rzadziej, są po prostu nazwami programów wykonujących zadanie — np. formatowanie dyskietki. Te polecenia nazywamy zewnętrznymi.

Oprócz wykonywania poleceń rezydentnych i uruchomienia programów, których nazwy podamy — przy czym nie ma żadnej różnicy między sposobem wydawania poleceń rezydentnych i zewnętrznych, różnica polega wyłącznie na sposobie ich realizacji — **command.com** potrafi jeszcze kilka rzeczy. Jedną z nich jest wykonywanie programów wsadowych. Taki program to zapisany na dysku ciąg poleceń (rezydentnych lub zewnętrznych), których nie musimy pracowicie wstukiwać z klawiatury za każdym razem, kiedy chcemy je wykonać. W programach wsadowych wolno używać kilku dodatkowych komend, nie mających sensu podczas pracy interakcyjnej, a pozwalających np. na testowanie rezultatów wykonanego polecenia i podejmowanie na podstawie wyniku testu prostych decyzji (np. jeśli nie udało się sformatowanie dyskietki, nie kopiuje się na nią zbioru danych). Drugą ważną cechą interpretera jest wykonywanie po włączeniu komputera programu wsadowego o nazwie **autoexec.bat**, co pozwala na automatyczne ustalenie warunków, w jakich będziemy pracować — na przykład na uruchomienie Norton Commander-a.

System operacyjny nie składa się wyłącznie z części opisanych powyżej — w jego skład wchodzi również wiele programów — poleceń zewnętrznych, realizujących różnorakie zadania. Właściwie poleceniem zewnętrznym jest każdy program użytkowy znajdujący się na dysku, ale termin ten stosuje się głównie do programów rozprowadzanych razem z omówionymi powyżej fragmentami DOS-u. Niektórymi poleceniami (wewnętrznymi i zewnętrznymi) zajmiemy się w najbliższym czasie w klanie.

Marcin Borkowski

KAT 90



Pracując na różnych komputerach, używając kilkudziesięciu lub więcej dyskietek różnych formatów na każdym z nich, musiałem zawsze poświęcać wiele godzin na utrzymanie porządku we wszystkich zbiorach, z których korzystałem. Niestety, mimo dużego nakładu pracy często znalezienie jakiegoś pliku było rzeczą kłopotliwą. Od dłuższego czasu nosiłem się z myślą napisania w miarę prostego i krótkiego programu, który ułatwiłby to zadanie.

Z bardzo prozaicznego powodu — trudność znalezienia wolnej chwili — pomysł nie został zrealizowany. Z tym większą ciekawością potraktowałem ofertę reklamową firmy AS Komputer dotyczącą programu KAT90. Propozycja przetestowania programu została przyjęta przychylnie, ale udostępnienie kopii zostało obwarowane wieloma zastrzeżeniami. W naszym kraju, gdzie prawa autorskie dotyczące własności programów są lekceważone nagminnie, firmy sprzedają oprogramowanie zabezpieczone w ten czy inny, ale nie zawsze skuteczny sposób.

Kat90 oferowany jest wyłącznie jako instalacja na dysku twardym użytkownika. Istotnym elementem transakcji jest instruktaż autorski, ponieważ programowi nie towarzyszy jakakolwiek dokumentacja użytkownika. Jedyną pomocą przy korzystaniu z programu są krótkie ekrany pomocy w języku polskim dostępne w każdej chwili po naciśnięciu klawisza F1. Normalna komunikacja z użytkownikiem prowadzona jest w języku angielskim. Mnie się to podoba, ponieważ polecenia są zwięzłe i bardziej zrozumiałe, ale niektórym użytkownikom, nie przepadającym za językiem ojczystym Williama Szekspira, może to sprawiać kłopot.

Program Kat90, którego kod źródłowy w języku C liczy parę tysięcy linii, umożliwia katalogowanie plików zapisanych na dyskietkach lub dyskach twardych użytkownika. Ponieważ kartoteka zbiorów przechowywana jest w trakcie pracy w pamięci operacyjnej komputera, liczba plików ograniczona jest do 7500—9000. Zaletą takiego rozwiązania jest duża szybkość programu.

Operacje menu głównego można podzielić na kilka grup. Do pierwszej należy zapis (SAVE) na dysk i odczyt (LOAD) bazy zawierające dane o plikach. Możliwe jest także łączenie (APPEND) wspomnianych baz. Druga grupa to operacje związane z wprowadzaniem (CREATE), modyfikacją (UPDATE) i usuwaniem (DELETE) informacji o poszczególnych dyskietkach. Do ostatniej grupy należą operacje pozwalające na przeglądanie bazy według możliwych kluczy i porządku oraz wyszukiwanie wskazanych plików: wybór zbiorów (SELECT), sortowanie (SORT), wyszukiwanie (FIND).

Kilkugodzinny instruktaż pozwala na opanowanie programu w sposób zadawalający. Jeśli przekonamy się do proponowanego przez autorów interfejsu użytkownika, praca z programem staje się w miarę prosta i przyjemna. Należy jednak przestrzegać pewnych ogólnych reguł porządku. Pracę zaczyna się od starannego ponumerowania wszystkich posiadanych dyskietek. Numeracja jest dwuczęściowa: numer pudełka z zakresu od 1 do 999 i numer dyskietki w pudełku — od 1 do 10.

Samo wprowadzanie danych do bazy jest dość proste i szybkie. Utworzenie kartoteki dla 41 dyskietek zajęło mi prawie 3 godziny. Duża część czasu poszła na wprowadzenie nowych etykiet zawierających numerację dyskietki. Mając zakończony ten proces można przystąpić do przeglądania bazy wg różnych zadanych kluczy i porządku. Dyskietki można posortować według rozmiaru wolnego miejsca, etykiety, numeru dysku oraz czasu i daty założenia etykiety. Klucze sortujące poszczególne pliki są następujące: nazwa, rozszerzenie, czas i data utworzenia, numer dysku. Generowane listy plików i inne raporty wysyłane są na ekran lub drukarkę.

Przestaje być problemem znalezienie dowolnego pliku, czy sprawdzenie na której dyskietce jest wystarczająco dużo miejsca na nowe programy. Łatwa jest także eliminacja kilku tych samych zbiorów, które znalazły się mniej lub bardziej przypadkowo na różnych dyskietkach.

Ponieważ bardziej odpowiada mi informacja na papierze niż prezentowana na ekranie, wydrukowałem wszystkie pliki o rozszerzeniach COM, EXE, PAS, CHI i ZIP. Dzięki temu udało mi się odnaleźć dawno „zagubione" pliki i stwierdzić, że mam co najmniej trzy różne zbiory COMMAND.COM na 7 różnych dyskietkach.

Mimo błędnej obsługi pewnych sytuacji krytycznych (brak dyskietki w napędzie lub zawirusowane pliki) i nie najlepiej wybranej przez autorów konwencji dialogu z użytkownikiem, do której długo musiałem się przekonywać, program podoba mi się i jest bardzo użytecznym, godnym polecenia narzędziem. Ma również dodatkową zaletę — uczy użytkownika metod utrzymania porządku i dyscypliny pracy. Obawiam się jednak, że z powodu dość wysokiej, jak dla prywatnego użytkownika, ceny oscylującej w pobliżu 700 tys. zł, znajdzie on głównie nabywców w grupie firm i instytucji, dysponujących taką kwotą.

Łatwość korzystania	2/5
Dokumentacja	1/5
Możliwości	5/5
Szybkość	5/5
Razem	13/20

MINUSY

- Niejasny intuicyjnie i kłopotliwy interfejs użytkownika
- błędna obsługa niektórych sytuacji krytycznych (brak dysku w napędzie, odłączona drukarka lub zawirusowana dyskietka)
- sprzedawany wyłącznie jako instalacja na dysku twardym użytkownika
- uboga dokumentacja dostępna tylko w formie ekranów pomocy

PLUSY

+ rozbudowana lista dostępnych operacji
+ duże możliwości różnorodnej selekcji przeglądanych plików i wyboru klucza sortującego
+ duża szybkość działania

Jarosław Młodzki

Opisywany program otrzymaliśmy od Pana Andrzeja Szarka, właściciela firmy AS Komputer, 03-920 Warszawa, ul. Walecznych 61, tel. 17-57-29, 13-04-11, tlx 816347 Ascom PL.

```
                         Empty base
JMS-1          14.50.10  09-07-1989    6144   001-01
BONDWELL-OS    13.33.26  22-07-1990       0   001-02
1-01-CHI       21.20.00  24-02-1991   19456   001-03
2-01-PRI01     21.23.58  24-02-1991    4096   002-01
2-02-PRI02     21.24.46  24-02-1991   83968   002-02
2-03-PRI03     21.25.32  24-02-1991  105472   002-03
2-04-ZXSIM     21.26.36  24-02-1991  142336   002-04
3-01-PLOT01    21.27.44  24-02-1991  187392   003-01
3-02-PLOT02    21.28.32  24-  ==== drive B: ====
3-03-PLOT03    21.29.10  24-  label  : 5-10-kniga4
3-04-PCW       21.29.52  24-  time   : 00.00.00
1-04-DOS01     21.31.36  24-  date   : 80.00.00
4-01-SLIDE     21.32.56  24-  space  :   123904
4-02-PBRUSH    21.33.46  24-
4-03-DOS02     21.35.00  24-02-1991   54272   004-03
4-04-DOS03     21.35.48  24-02-1991  254976   004-04
5-01-KNIGA1    21.38.38  24-02-1991  157696   005-01
5-03-KNIGA2    21.39.16  24-02-1991  148480   005-03
5-05-KNIGA3    21.39.52  24-02-1991  163840   005-05
(null)                               123904   005-06
```

Rys. 2 Wprowadzanie nowych dyskietek

```
               AS KOMPUTER Co. Ltd

           professional  software
            marketing & service

                 K A T - 9 0

              WITH COMPLIMENTS
                A U T H O R S

      Wojciech Guzinski, Andrzej Szarek

Head-Office in Poland          Branch Office in Poland
Walecznych 61                      A.W.Kowalskiego 7
03-920 Warszawa                       16-400 Suwalki
tel. 17-57-29                          tel. 61-2   38
tlx 81-6347 askom pl            tlx 052-2114 asko   1

Wlascicielem kopii jest :  Jaroslaw MLODZKI - Warszawa
```

Rys. 1 Wizytówka programu

```
                    SOrt
SOrt - sortowanie:                                2-03 : \SWEDEN\
Sortowanie ma dwie mozliwosci:                    2-03 : \LETTERS\
                                                  2-03 : \LETTERS\
- Disk - sortowanie etykiet dyskietek po dowolnej 2-03 : \LETTERS\
sekwencji nastepujacych parametrow etykiety dys-  2-03 : \ANGLIA\OLD\
kietki:                                           2-03 : \LETTERS\
 Size   -ilosc wolnego miejsca na dyskietce ros-  2-03 : \ANGLIA\
         naco,                                    2-03 : \LETTERS\
 Label  -etykieta dyskietki po odrzuceniu infor-  2-03 : \LETTERS\
                                         more..   2-03 : \RFN\        Menu
                                                  2-03 : \TAIWAN\     LOad
 456 OWEN     .LTR  01.43.54  16-11-1987   1468   002-03 : \ANGLIA\   SAve
 457 OWEN2    .LTR  16.22.46  07-01-1987   1098   002-03 : \ANGLIA\   CReate
 458 RAINER   .LTR  16.10.18  05-11-1987   1004   002-03 : \RFN\      Update
 459 SECHMAN  .LTR  14.56.00  02-09-1987   2001   002-03 : \KOMP\     Find
 460 KL-READ  .ME   22.15.08  23-12-1989    538   003-02 : \IBM\      SElect
 461 KL-READ  .ME   22.15.08  23-12-1989    538   003-03 : \KLAUS\    Delete
 462 READ     .ME   08.23.58  13-11-1985   4222   001-03 : \          SOrt
 463 TURBO    .MSG  03.33.20  01-03-1985   1536   001-01 : \          LIst
 464 CONFIG   .PAR  09.33.04  28-10-1986     72   001-03 : \          Reject
                                                                      Inv-al
F1 - help            AltF1 - DOS shell      CtrlF1 - program-info     All-se
```

Rys. 3 Przeglądanie plików posortowanych wg zadanego klucza





DRUKARKA

D-100 MPC

W chwili gdy polski rynek komputerowy zdominowały produkty z zagranicy, warto poświęcić nieco uwagi każdej próbie rodzimego przemysłu. Co prawda produkcja polskich komputerów liczy się ledwie w setkach egzemplarzy, ale nasi wytwórcy starają się czasem nas zaskoczyć. Kilka lat temu taką rewelacją wśród krajów Europy Wschodniej była drukarka D-100. Jej producent — firma Mera Błonie, modernizując stale swój produkt, wypuściła ostatnio na rynek nowe modele tej drukarki.

D-100 MPC jest 9-igłową drukarką mozaikową średniej klasy, przeznaczoną do współpracy z 8-bitowymi komputerami domowymi, jak również 16-bitowymi klasy IBM PC XT/AT. Stąd też wyniknął podział testu jej działania na dwie części — test współpracy **D-100** z 8-bitowym ZX-Spectrum i 16-bitowym IBM PC/AT.

Budowa

Pod względem konstrukcyjnym testowany model odbiega nie tylko od swego pierwowzoru, ale od wszystkich znanych nam popularnych drukarek mozaikowych.

Można w nim stosować papier w pojedynczych kartkach, a po dołączeniu z tyłu drukarki zewnętrznego podajnika papieru (traktora), także papier z perforacją lub z rolki. Podajnik ten został źle pomyślany i jego użycie jest bardzo niewygodne. Aby założyć papier do drukarki, należy zdjąć podajnik, częściowo go rozłożyć, przełożyć przezeń papier, wsunąć papier do drukarki, złożyć i zamontować podajnik z tyłu drukarki.

Dużo wygodniejsze jest korzystanie z pojedynczych arkuszy, a to ze względu na ciekawie skonstruowany podajnik do arkuszy formatu A4. Przy korzystaniu z papieru w rolce można podajnik położyć i jego ostrą krawędź używać do oddzierania papieru.

Problematyczne jest stosowanie papieru o mniejszym formacie arkuszy bądź węższego niż 8 cali, ze względu na mechanikę drukarki. Brak jest bowiem rolek dociskających zadrukowany papier do wałka. Razem z rolkami wyeliminowano podziałkę w calach (bądź centymetrach) ułatwiającą ocenę szerokości tekstu. Przy wąskim papierze głowica powracając od prawego marginesu napotyka na krawędź papieru przedzierając go na pół. Tak drastyczne uproszczenie mechaniki miało zapewne na celu zwiększenie jej niezawodności, ale za to znacznie obniżyło komfort pracy.

Producent zastosował krajową głowicę dość dużych rozmiarów, o żywotności rzędu 100 mln znaków ale bardzo hałaśliwą. Głowica jest przesuwana paskiem zębatym, jej ruchy są częściowo optymalizowane i teksty drukowane są dwukierunkowo. Dla druku dwukierunkowego konieczne jest wstępne ustawienie luzów powrotnego ruchu głowicy, tak aby tekst w kolejnych liniach nie był przesunięty w fazie. Dokładne ustawienie za pomocą 4 mikroprzełączników) jest bardzo trudne i może trwać ponad godzinę.

Do drukowania używana jest zwykła taśma szer. 13 mm na szpulkach, tania i łatwa do kupienia. Jakość druku dla takiej taśmy jest dobra zarówno w trybie draft, jak i w trybie korespondencyjnym (NLQ), w który zostały wyposażone oba testowane modele. W wydruku można otrzymać oryginał i dwie czytelne kopie (przez kalkę). Łączna grubość papieru nie może być większa od 0.3—0.5 mm, jeśli chcemy, aby nie blokował się wysuw papieru.

W trybie draft prędkość druku wynosi ok. 100 zn./sek., a w trybie NLQ prędkość spada do 30 zn.

Wiele do życzenia przedstawia obudowa. O ile jej spód jest wykonany z grubej blachy i stanowi dobrą podstawę dla mechaniki drukarki, to góra jest wykonana z cienkiego plastiku. Tworzywo to łatwo się brudzi, a trudno go czyścić ze względu na chropowatą powierzchnię. Plastikowa forma jest krzywo ukształtowana i zbyt delikatna, skoro ugina się do środka przy wciskaniu sensorowych klawiszy na pulpicie. Ponadto z niewiadomych przyczyn gałka do wysuwu papieru została umieszczona po lewej stronie obudowy, zamiast po prawej.

W odróżnieniu od części mechanicznej, pełnej nowatorskich pomysłów, część elektroniczna sprawia wrażenie koncepcji dobrze przemyślanej i zrealizowanej. Na płytce drukowanej panuje porządek, nie ma prowizorycznych rozwiązań, co zapewnia dużą niezawodność. Wśród wielu układów scalonych, oprócz procesora, zamontowano 8KB pamięci RAM, której 2 do 5KB (zależnie od trybu) wykorzystuje się na bufor wczytywanych znaków, oraz tzw. download font — zestaw znaków definiowanych przez użytkownika. Z tej ostatniej możliwości korzysta się, aby definiować polskie litery. W tym przypadku polskich znaków nie trzeba definiować, pośród bowiem kilkunastu zestawów znaków, jakie **D-100 MPC** oferuje, można wybrać układ znaków **IBM PC** w wersji z polskimi znakami w tzw. standardzie „Mazovii" (patrz tabela). Oprócz tego producent oferuje zestawy znaków z cyrylicą lub w standardzie Latin 2.

Mimo tych udogodnień lepiej zdefiniować własne polskie znaki, bo fabryczne są wyraźnie niedopracowane. Jedynie w trybie NLQ znaki te zachowują proporcje.

D-100 MPC wyposażona jest w złącze typu CENTRONICS (zastosowano niestandardowe, 37-stykowe gniazdo CANON DB 37). Do gniazda tego można dołączyć dodatkowy, oddzielnie zasilany adapter interfejsu szeregowego V-24.

Drukarka została dobrze oprogramowana i daje możliwość stosowania czterech krojów pisma: Pica (10 pkt./cal, 80 zn. w linii), Elite (12, 96), Condensed (17, 136) oraz Proportional. Każdy z nich może być dodatkowo modyfikowany: podkreślony, pochyły (kursywa), podwójny bądź fazowy. Niektóre z tych opcji są też dostępne w trybie NLQ oraz dla druku Subscript i Superscript. Zależnie od ustawienia mikroprzełączników i wysłanych kodów sterujących można drukować znaki w standardzie **IBM** (plus znaki polskie), **EPSON** albo w trybie pośrednim.

Dla użytkownika jest dostępna większość kodów sterujących i trybów drukowania tekstów oraz grafiki, którymi dysponują inne drukarki tej klasy, np.: drukarki firm **EPSON** i **STAR**. Tak więc można wybrać druk jedno- i dwukierunkowy, różne kroje pisma, grafikę pojedynczej, podwójnej i poczwórnej gęstości, justowanie tekstu, ustawianie marginesów i długości stron itd.

Znaki drukowane są w matrycy 9 * 12 pkt. dla druku draft oraz 18*24 dla trybu NLQ. W trybie graficznym można uzyskać 60, 72, 80, 90, 120 i 240 pkt./cal.

Do sterowania drukarką, oprócz kodów sterujących, służą sensory umieszczone wraz z diodami świecącymi po prawej stronie obudowy. Sensory działały pewnie, ale ich konstrukcja jest bardzo delikatna. Ręcznie można przestawić drukarkę w stan ON/OFF LINE, wysunąć papier o wiersz (LF) albo o stronę (FF), włączyć tryb korespondencyjny (NLQ).

Włączenie drukarki z wciśniętym klawiszem LF wywołuje test drukarki, a z klawiszem FF przestawia ją w tryb **HEX DUMP** kopiowania w dwóch kolumnach: kodów szestnastkowych docierających do drukarki oraz odpowiadających im kodów ASCII. Poprzez skomplikowane operacje sensorami ON LINE, LF i FF można wybrać dowolny krój druku, podkreślenie znaków itp., ale nie można ustawić marginesów. Diody świecące sygnalizują stan **ON LINE**, tryb **NLQ**, gotowość drukarki (READY) oraz brak papieru (PAPER OUT). Bardzo brakuje diod sygnalizujących tryb druku, bo nie wiadomo, jaki krój druku ustawił się w drukarce po kilkukrotnym wciśnięciu sensorów.

Oprócz diod drukarka sygnalizuje swój stan brzęczykiem, którego lubi nadużywać. Na przykład brak papieru jest sygnalizowany piskami trwającymi

Test polskich znaków w standardzie "Mazovii" dla drukarki D100 MPC

ą - 134	Ą - 143	ó - 162	Ó - 163
ć - 141	Ć - 149	ś - 158	Ś - 152
ę - 145	Ę - 144	ż - 166	Ż - 160
ł - 146	Ł - 156	ź - 167	Ź - 161
ń - 164	Ń - 165	zł - 155	

Test polskich znaków w standardzie "Mazovii" dla drukarki D100 MPC

ą - 134	Ą - 143	ó - 162	Ó - 163
ć - 141	Ć - 149	ś - 158	Ś - 152
ę - 145	Ę - 144	ż - 166	Ż - 160
ł - 146	Ł - 156	ź - 167	Ź - 161
ń - 164	Ń - 165	zł - 155	

ponad 5 sek. W tym czasie drukarka na nic nie reaguje, nawet na nową stronę papieru.

Dodatkowo czujnik braku papieru zbyt wcześnie przerywa drukowanie, gdy można jeszcze wydrukować 10—15 wierszy tekstu.

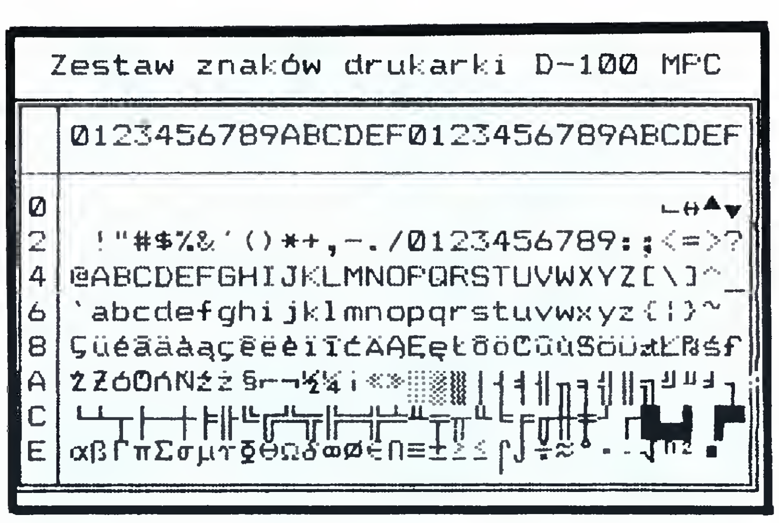
Pierwsze wrażenia

D-100 MPC otrzymała wygodne i mocne opakowanie, co wróżyło, że drukarka wytrzyma do końca testu. Jednak obejrzenie zawartości pudła szybko zmieniło naszą opinię (obudowa). Po przeczytaniu instrukcji doszliśmy do wniosku, że producent przekazał nam dokumentację fabryczną bądź serwisową. Później dowiedzieliśmy się, że znów jesteśmy w błędzie i dwie książki formatu A4, liczące łącznie ponad 150 stron, odbite na ksero to właśnie jest instrukcja. Nie zrażeni pomyłką, założyliśmy taśmę, traktor, papier i postawiliśmy drukarkę na stole. Okazało się, że wkręty wystające od spodu obudowy są dłuższe od gumowych nóżek i rysują stół. Dokręciliśmy je. Włączyliśmy drukarkę i stwierdziliśmy, że w czasie druku nie przesuwa się taśma barwiąca. Znów konieczny okazał się śrubokręt. W tym momencie mieliśmy już dość testowania, jednak zwyciężyła ciekawość.

Eksploatacja

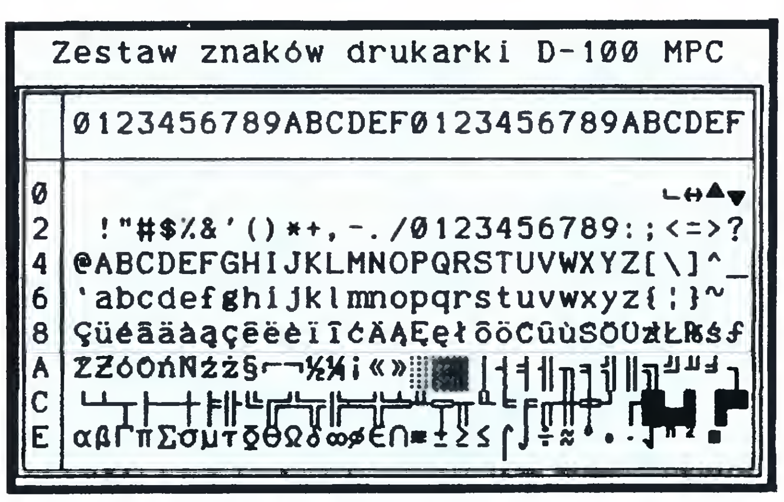
Podłączyliśmy drukarkę, za pomocą interfejsu **MICROFACE CENTRONICS**, do **ZX Spectrum**. Używając programu CITO PRINT drukowaliśmy teksty spod Tekst-Edytora. Programy w BASIC-u, wydruki spod GENS-a i MONS-a uzyskaliśmy korzystając z pomocy programu MICROFACE, a obrazki za pomocą poprawionego ART STUDIO. Nasze spostrzeżenia można podsumować w kilku punktach:

— przy drukowaniu tekstów drukarka lubiła śmiecić kodami kontrolnymi. Wszystkie kody były wysyłane w standardzie Epsona. Drukarka powinna je bez problemu przyjąć. Czasem, z niewiadomych powodów, przełączały się kroje znaków i fonty.
— programy w BASIC-u drukowane były bez problemów
— wydruki asemblerowe również były zadowalające
— najgorzej drukowały się ekrany. Gubione bywały linie (czasem wyłączała się jedna igła), rysunki były nierówne w pionie mimo korekcji luzów za pomocą mikroprzełączników pod pokrywą, za to nie występowały białe smugi pomiędzy liniami wydruku, co spotykane jest w wielu drukarkach.


Zestaw znaków drukarki D-100 MPC

0123456789ABCDEF0123456789ABCDEF

2 !"#$%&'()*+,-./0123456789:;<=>?
4 @ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_
6 `abcdefghijklmnopqrstuvwxyz{|}~


Zestaw znaków drukarki D-100 MPC

0123456789ABCDEF0123456789ABCDEF

2 !"#$%&'()*+,-./0123456789:;<=>?
4 @ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_
6 `abcdefghijklmnopqrstuvwxyz{|}~

— jeden z trybów graficznych, wybierany kodami **ESC,"*",5,** służący do wydruków proporcjonalnych (72 pkt./cal w obu osiach) ma błąd uniemożliwiający jego wykorzystanie.

Drukarka często zawieszała się (nie reagowała na dane, świeciły się wszystkie diody jednocześnie: **ON LINE**, **READY**, **NLQ** i **PAPER OUT!**). Źle skonstruowano zasilacz drukarki: jest duży i ciężki. Potrafił on, po wyjęciu wtyczki z gniazdka, podać na bolce wtyczki taki ładunek, że kontakt ręki z nim nie należał do przyjemnych. Bardzo twardy i nieelastyczny kabel łączący drukarkę z komputerem dostarczony przez producenta przeszkadzał przy pracy: Spectrum i stacja Timex jeździły po stole za drukarką. Ten sam kabel winny był większości przekłamań w wydrukach, ponieważ powodował wysuwanie się wtyczek z gniazda drukarki i interfejsu komputera.

Podczas testu **D-100 MPC** z **IBM PC** drukarka wykazała sporo zalet, właściwie nie było programu, którego nie można było nakłonić do współpracy z nią. Próbowaliśmy kopiować ekran, drukować pliki tekstowe, obwody elektroniczne z OrCAD-a, rysunki z Print Mastera, Dr Halo itp. Właściwie jedynym problemem było znalezienie takiego drivera w programie, który by wysyłał kody kontrolne akceptowane w całości przez **D-100**. Z reguły wybieraliśmy driver **EPSON FX/MX** albo **IBM GRAPHICS PRINTER**. Przeszkadzał jedynie zbyt mały bufor znakowy, gdyż trzeba było czekać na zakończenie drukowania, aby testować nowy program. PRINT.EXE — rezydentny program DOS-u buforujący wydruki nie zdał egzaminu. Bardzo przydatne okazały się polskie litery w standardzie „Mazovii", tym bardziej, że i nasza karta graficzna była w nie zaopatrzona. W takiej konfiguracji, korzystając dodatkowo z programu POLKLAW, można było bez problemu pisać i drukować po polsku.

Do nieprawidłowych objawów należy zaliczyć wieszanie się drukarki przy wyłączonym IBM PC — nie można było nawet uruchomić auto-testu drukarki.

Podsumowanie

Dobra współpraca z IBM PC z użyciem polskich znaków oraz oznaczenie drukarki (MPC) skłania do wniosku, że drukarka została wyprodukowana z myślą o podłączeniu jej do „Mazovii". Dołączenie jej do innych komputerów wiąże się z szukaniem programu, który potrafiłby z nią współpracować, albo z przeróbką istniejących programów. Generalnie jest to urządzenie ciekawe, zaprojektowane z rozmachem, o bogatych możliwościach. Jedynie jego wykonanie może budzić poważne zastrzeżenia i w połączeniu z wygórowaną ceną (ok. 3 mln zł) może stać się przyczyną małego zainteresowania klientów. Mimo dostępności tej drukarki w sklepach Składnicy Harcerskiej i sklepie firmowym w Błoniu profesjonaliści zapewne wybiorą bardziej niezawodne modele drukarek, a prywatni użytkownicy poszukają tańszych ofert.

Obniżenie ceny, zmiana obudowy i zastosowanie lepszego zasilacza wydają się być konieczne, aby drukarka D-100 MPC mogła odnieść sukces na naszym rynku.

ZALETY

— bogaty zestaw trybów druku, krojów liter i fontów
— pełny wybór kodów sterujących standardu EPSON
— polskie litery w standardzie „Mazovii" lub Latin 2
— dobrze opracowany tryb korespondencyjny (NLQ)
— możliwość wyboru papieru w arkuszach, w rolce i z perforacją
— różnorodne warianty formatowania i justowania wydruków
— stosowanie łatwo dostępnej taśmy
— opcja **HEX DUMP** drukuje także znaki **ASCII**

WADY

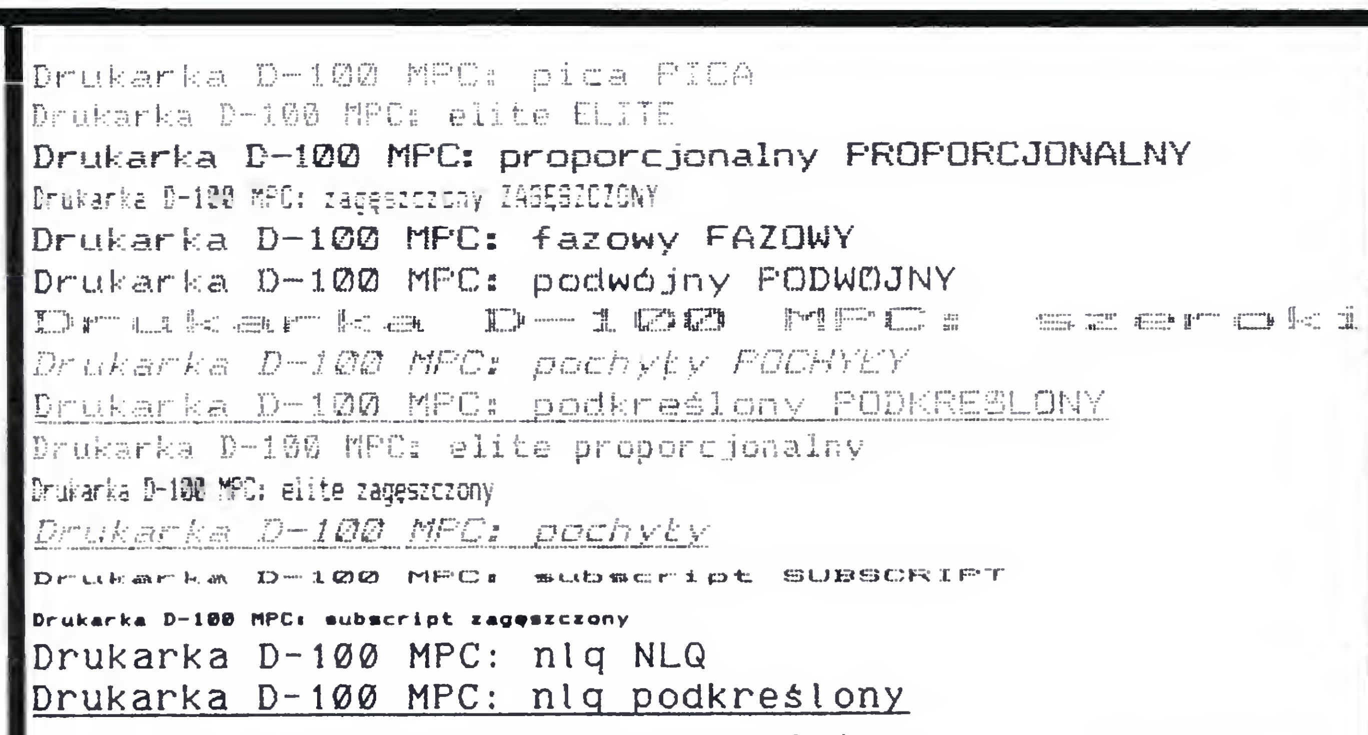
— niebezpieczny zasilacz
— niestandardowe złącze **CENTRONICS**
— toporna obudowa
— „nowatorska" mechanika podawania i prowadzenia papieru
— trudno dostępny wyłącznik sieciowy
— źle wykonany kabel połączeniowy
— „gadatliwość" i tendencje do wieszania się
— mała pojemność bufora w trybie IBM
— niechlujna instrukcja
— wysoka cena

Marek Sawicki
Grzegorz Ostapiuk

Parametry techniczne drukarki D-100 MPC na podstawie instrukcji

głowica	— 9 igieł, 100 mln znaków
szybkość druku	— 100—120 zn./sek. w trybie draft, 30 zn./sek w NLQ
matryca znaków	— 9x12 tryb draft 18x24 tryb NLQ
wielkość znaku	— 2,1 mm x 2,5 mm dla druku Pica 10 zn./cal
gęstość druku	— Pica 10 zn./cal Elite 12 zn./cal Condensed 17 zn./cal Elite Condensed 20 zn./cal
odstęp wierszy	— 1/6, 1/8, n/72, n/216 cala
bufor	— 2 KB, 5 KB w trybie IBM
papier	— arkusze A4, perforowany 100 do 250 mm, rolkowy do 210 mm
interfejs	— równoległy CENTRONICS, 8-bit, złącze 37 stykowe
zasilanie	— 220V (+10%, −15%), 50/60 Hz 35 VA, kabel 1,2 m
wymiary	— 405x80x300 mm
masa	— ok. 6 kg
cena	— ok. 3 mln zł

Dziękujemy Zakładom Mechaniczno-Precyzyjnym „MERA-BŁONIE", ul. Grodziska 15, za udostępnienie drukarki do testu.


Drukarka D-100 MPC: pica PICA
Drukarka D-100 MPC: elite ELITE
Drukarka D-100 MPC: proporcjonalny PROPORCJONALNY
Drukarka D-100 MPC: zagęszczony ZAGĘSZCZONY
Drukarka D-100 MPC: fazowy FAZOWY
Drukarka D-100 MPC: podwójny PODWÓJNY
Drukarka D-100 MPC: szeroki
Drukarka D-100 MPC: pochyły POCHYŁY
Drukarka D-100 MPC: podkreślony PODKREŚLONY
Drukarka D-100 MPC: elite proporcjonalny
Drukarka D-100 MPC: elite zagęszczony
Drukarka D-100 MPC: pochyły
Drukarka D-100 MPC: subscript SUBSCRIPT
Drukarka D-100 MPC: subscript zagęszczony
Drukarka D-100 MPC: nlq NLQ
Drukarka D-100 MPC: nlq podkreślony
Drukarka D-100 MPC: nlq subscript

TEST !



MONITOR—TELEWIZOR

BIAZET TMP 205

Zgodnie z zapowiedzią prezentujemy redakcyjny test telewizora-monitora, który otrzymaliśmy od zakładów UNITRA BIAZET w Białymstoku. Dziękujemy.

Połączenie telewizora i monitora w jedną funkcjonalną całość jest wygodnym rozwiązaniem i zwiększa uniwersalność jego zastosowań. Jest to właściwie fabryczna realizacja starego pomysłu wielu majsterkowiczów. W niniejszym opisie ograniczę się do testu samego monitora, co zapewne najbardziej interesuje Czytelników.

Telemonitor został umieszczony w estetycznej i funkcjonalnej obudowie z białego tworzywa, identycznej jak w prezentowanym już monitorze MMK 127. Zachowano takie samo rozmieszczenie i opis podstawowych pokręteł regulacyjnych i gniazd przyłączeniowych. Na płycie czołowej znalazły się również regulatory telewizora, jak na przykład przełącznik kanałów.

Dla użytkownika komputera najważniejszy przełącznik zamieniający telewizor w monitor umieszczono na tylnej ściance w pobliżu gniazda podłączenia mikrokomputera. Takie umiejscowienie jest dość niewygodne, szczególnie przy konieczności częstej zmiany trybu pracy telewizor — monitor.

Wyświetlany obraz jest czarno-biały. Jego jakość nie jest już najwyższa, powodem jest prawdopodobnie inny typ kineskopu, przeznaczony do odbiorników TV. Dla popularnych komputerów domowych nie ma to większego znaczenia, jednak z kartą CGA i IMB PC jest już dokuczliwe. Aby się przekonać jaka jest faktyczna rozdzielczość monitora, zajrzałem do instrukcji obsługi. Okazało się, że załączona do telemonitora instrukcja nie jest od niego, lecz od telewizora TMP 201!

Układy elektryczne nie odtwarzają składowej stałej sygnału wizji. Zmusza to do częstych regulacji jasności i kontrastu. Jeśli przy ciemnym tle, skokowo wyświetlimy duży jasny obrazek, to początkowo przez krótką chwilę jego jasność będzie duża, a następnie zmaleje do wartości ustalonej.

Zniekształcenia geometryczne obrazu są niezauważalne. Podobnie jak w prezentowanym już monitorze tekst rozmywa się nieco w rogach ekranu, jednak dokuczliwe jest to dopiero przy bardzo dużej jasności obrazu.

Podobieństwo przedstawionych zjawisk w telemonitorze i monitorze MMK 127 nie jest zapewne przypadkowe. Zapewne układy elektryczne i podzespoły w obu urządzeniach są identyczne. Można się o tym dodatkowo przekonać, jeśli podłączymy telemonitor do karty CGA. Oczywiście znów konieczne było rozkręcenie obudowy i regulacja synchronizacji pionowej, aby możliwa była współpraca. Komputery 8-bitowe współpracowały z telemonitorem poprawnie.

W redakcji telemonitor pracował ponad dwa miesiące. W tym okresie nie wystąpiły żadne usterki w jego pracy. Na zakończenie oczywiście należy wymienić jego zalety i wady:

ZALETY:

1. Estetyczna i funkcjonalna obudowa.
2. Łatwo dostępne i czytelnie opisane podstawowe pokrętła regulacyjne.

WADY:

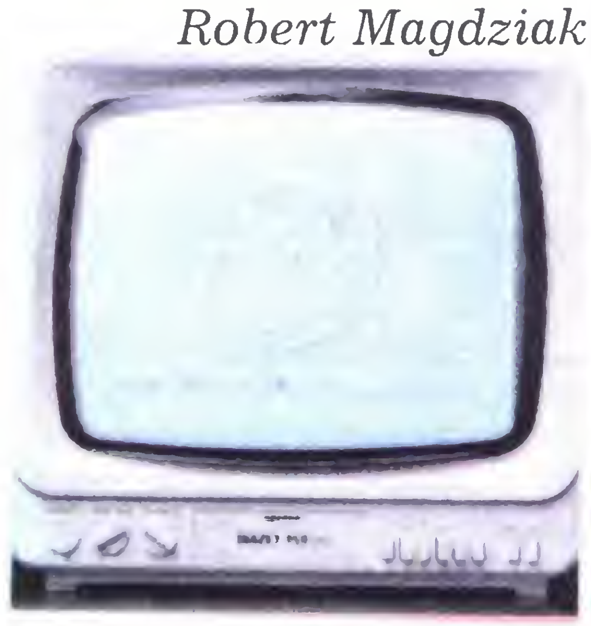
1. Trudno dostępny przełącznik telewizor-monitor.
2. Brak układu odtwarzania składowej stałej sygnału wizji.
3. Zła współpraca z kartą CGA.

Parametry techniczne:

Kineskop	12" czarno-biały
Częstotliwość odchylania poziomego	15625 Hz
Częstotliwość odchylania pionowego	50 Hz
Maksymalna moc fonii	1 W
Moc pobierana z sieci	35 W
Wymiary	312*330*272 mm
Masa	6.8 kg

Telemonitor można kupić w sklepie fabrycznym zakładów BIAZET w Białymstoku, ul. Sz. Płn. Obwodowa 38 w cenie 840 tys. zł.

Robert Magdziak



ZRÓB TO SAM

8255 — okno na świat
SILNIK KROKOWY

Każdy, kto próbował przy pomocy komputera sterować zwykłym silnikiem elektrycznym wie, jak trudno dokładnie kontrolować ilość i szybkość jego obrotów.

Tymczasem istnieje pewien rodzaj silników wprost stworzonych do sterowania cyfrowego — silniki krokowe. W sprzęcie komputerowym stosuje się je w drukarkach (przesuw papieru i głowicy drukującej), ploterach (przesuw pisaka) i w stacjach dysków (przesuw głowicy nad dyskiem).

Jak sugeruje sama nazwa, obroty tych silników następują skokami. Pojedynczy krok (kąt obrotu) wynosi w zależności od typu silnika i wykonania od 45 stopni do 0,72 stopnia. Oznacza to, że w pierwszym przypadku do wykonania jednego pełnego obrotu trzeba wykonać 8 kroków, a w drugim przypadku 500 kroków! Aby silnik krokowy wykonał jeden krok, trzeba dołączyć napięcie do odpowiednich uzwojeń (faz) silnika w ściśle określony sposób. Każdy komputer przez odpowiedni interfejs poradzi sobie z tym doskonale.

ZASADA DZIAŁANIA

Rys. 1 przedstawia schematyczny model magnetoelektrycznego, unipolarnego czterofazowego silnika krokowego. Rys. 2 przedstawia schemat ideowy tego silnika wraz z kluczami (przełącznikami) sterującymi. Popatrzmy na rys. 1. W środku znajduje się wirnik z zaznaczonymi biegunami magnetycznymi. Wokół wirnika symetrycznie rozmieszczone są uzwojenia stojana oznaczone L1—L4. Jeden przewód każdego uzwojenia połączony jest z pozostałymi i z plusem napięcia zasilania. Napięcie to jest charakterystyczne dla każdego typu silnika, najczęściej jest to 5 lub 12 Volt. Drugi przewód jest zwierany do masy (minus zasilania) przez klucz elektroniczny oznaczony K1—K4 na rys. 2.

Wyobrażmy sobie że włączamy klucz K1. Przez uzwojenia L1 popłynie prąd. Część stojana związana z L1 staje się elektromagnesem i zaczyna przyciągać biegun N wirnika (Czy na pewno? Przypomnij sobie wiadomości z elektromagnetyzmu i sprawdź!). I nic się nie dzieje! Pozornie. Gdybyśmy teraz spróbowali obrócić wirnik to nie będzie to łatwe. Odłączmy klucz K1 i włączmy klucz K2. Natychmiast wirnik wykona obrót o 90 stopni w prawo gdyż biegun N zostanie przyciągnięty przez L2. Odłączmy klucz K2 i włączmy K3. Wirnik znowu obróci się o 90 stopni w prawo gdyż biegun N zostanie przyciągnięty przez L3. Wyłączenie klucza K3 i włączenie klucza K4 spowoduje wykonanie kolejnego kroku o 90 stopni w prawo. Wyłączenie K4 i włączenie K1 to kolejny krok o 90 stopni w prawo. W ten sposób wirnik wykonał 4 krokami pełen obrót. Kolejność włączania kluczy była następująca: K1, K2, K3, K4, K1, K2, K3, K4, K1...

Taka sekwencja włączania kluczy powodowała obroty w prawo. A obroty w lewo? Nic prostszego! K1, K4, K3, K2, K1, K4, K3, K2, K1... A jakie kroki i w którą stronę wykona wirnik sterowany następującą sekwencją (początkowo wirnik ustawiony jak na rys. 1): K2, K1, K2, K1, K4, K1?

Wydaje się, że minimalny możliwy do uzyskania krok wynosi 90 stopni. Spójrzmy raz jeszcze na rys. 1. Włączamy K1 i nic się nie dzieje. Teraz nie wyłączając K1 włączamy K2. Co się stało? Biegun N wirnika został przyciągnięty między L1 a L2. Wirnik wykonał pół kroku o 45 stopni w prawo. Teraz wyłączamy K1 co powoduje kolejne pół kroku o 45 stopni w prawo. Włączenie K3 to kolejne pół kroku o 45 stopni w prawo, wyłączenie K2 to kolejne pół kroku...

Sekwencja włączania kluczy dla obrotów w prawo przy pracy półkrokowej jest następująca: K1, K1K2, K2, K2K3, K3, K3K4, K4, K4K1, K1... W lewo: K1, K1K4, K4, K4K3, K3, K3K2, K2, K2K1, K1... Jak widać przy takim sterowaniu na pełny obrót potrzeba 8 półkroków, a jeden krok wynosi 45 stopni.

LISTING 1

```
   5 LET RS=223: LET PortA=31: L
ET WT=1
   8 OUT RS,BIN 10000000
  10 DIM T(4)
  20 LET T(1)=BIN 00000001
  30 LET T(2)=BIN 00000010
  40 LET T(3)=BIN 00000100
  50 LET T(4)=BIN 00001000
  57 GO TO 300
  59 REM Obroty w prawo
  60 FOR I=1 TO ILE
  70 IF WT=5 THEN LET WT=1
  80 OUT PortA,T(WT)
  90 LET WT=WT+1
 100 NEXT I
 110 RETURN
 150 REM Obroty w lewo
 160 FOR I=1 TO ILE
 170 IF WT=0 THEN LET WT=4
 180 OUT PortA,T(WT)
 190 LET WT=WT-1
 200 NEXT I
 210 RETURN
 300 INPUT "Obroty w prawo-[illegible]
                    w lewo-[illegible]
   ";a$
 310 INPUT "Ile krokow ?      ";I
LE
 320 IF a$="p" OR a$="P" THEN GO
TO 400
 330 IF a$="l" OR a$="L" THEN GO
TO 500
 340 GO TO 300
 400 GO SUB 60: GO TO 300
 500 GO SUB 160: GO TO 300
```

LISTING 2

```
   30 ;Przyklad sterowania silni
ka krokowego  (o) RGB
   40 ;
   50 PORTA   EQU  31
   60 PORTB   EQU  95
   70 PORTC   EQU  159
   80 CWR     EQU  223
   90 ;
  100         ORG  60000
  110 ;
  120 ILEKRO  DEFW 0
  130 KIER    DEFB 0 ;0=Lewo,1=Pr
awo
  140 CZAS    DEFW 500
  150 WSKTBX  DEFW POTAB
  160 ;
  170 INIT    LD   A,%10000000 ;W
szystkie porty WY
  180         OUT  (CWR),A
  190         RET
  200 ;
  210 ;TABLICA KROKOW SILNIKA
  220 ;
  230 POTAB   DEFB %00000001
  240         DEFB %00000011
  250         DEFB %00000010
  260         DEFB %00000110
  270         DEFB %00000100
  280         DEFB %00001100
  290         DEFB %00001000
  300         DEFB %00001001
  310 ;
  320 ;Modyfikacja WSKaznika TaB
licy krokow w zaleznosci od kier
unku obrotow
  330 ;
  340 XLEWO   LD   HL,(WSKTBX)
  350         DEC  HL
  360         LD   (WSKTBX),HL
  370         LD   DE,POTAB-1
  380         AND  A
  390         SBC  HL,DE ;CZY ZMI
ENIC WSKAZNIK TABLICY KROKOW
  400         JR   NZ,BUWYLX
  410         LD   HL,POTAB+7 ; 8
 KROK
  420         LD   (WSKTBX),HL
  430 BUWYLX  RET
  440 ;
  450 ;
  460 XPRAWO  LD   HL,(WSKTBX)
  470         INC  HL
  480         LD   (WSKTBX),HL
  490         LD   DE,POTAB+8
  500         AND  A
  510         SBC  HL,DE ;CZY ZMI
ENIC WSKAZNIK TABLICY KROKOW
  520         JR   NZ,BUWYPX
  530         LD   HL,POTAB
  540         LD   (WSKTBX),HL
  550 BUWYPX  RET
  560 ;
  570 ;Wyslanie bajtu z tablicy
krokow wskazanego przez wskaznik
  580 ;
  590 WYBUF   CALL OPOZN
  600         LD   HL,(WSKTBX)
  610         LD   A,(HL)
  620         OUT  (PORTA),A
  630         RET
  640 ;
  650 ;Petla czasowa realizujaca
 opoznienie miedzy krokami
  660 ;
  670 OPOZN   PUSH BC
  680         LD   BC,(CZAS)
  690 OP1     DEC  BC
  700         LD   A,B
  710         OR   C
  720         JR   NZ,OP1
  730         POP  BC
  740         RET
  750 ;
  760 ;Glowna procedura wykonuja
ca zadana liczbe krokow
  770 ;
  780 OBROTY  PUSH BC
  790         LD   BC,(ILEKRO)
  800 OB1     CALL SPRAW
  810         DEC  BC
  820         LD   A,B
  830         OR   C
  840         JR   NZ,OB1
  850         POP  BC
  860         RET
  870 ;
  880 ;Rozpoznawanie kierunku ob
rotow i wysylanie bajtow
  890 ;sterujacych do portu
  900 ;
  910 SPRAW   LD   A,(KIER)
  920         CP   0
  930         JR   Z,RAZLEW
  940 RAZPRA  CALL XPRAWO
  950         CALL WYBUF
  960         RET
  970 RAZLEW  CALL XLEWO
  980         CALL WYBUF
  990         RET
```

LISTING 3

```
  10 LET LICZBA=0: LET STRONA=0:
LET OPOZN=500
  20                REM
 100 LET LICZBA=200
 110 LET STRONA=1  : REM W prawo
 120 LET OPOZN=300
 130 GO SUB 9000
 140 STOP
8990                REM
9000 POKE 60000,LICZBA-256*INT (
LICZBA/256): POKE 60001,INT (LIC
ZBA/256)
9010 POKE 60002,STRONA
9020 POKE 60003,OPOZN-256*INT (O
POZN/256): POKE 60004,INT (OPOZN
/256)
9030 RANDOMIZE USR 60086
9040 RETURN
```

W rzeczywistym silniku opisywanego typu elektromagnesy każdej fazy powtarzają się co pięć w następującej kolejności: L1, L2, L3, L4, L1, L2, L3, L4, L1... Zakładając że istnieje 25 takich czwórek uzyskujemy silnik 100 (25*4) krokowy, lub 200 krokowy przy sterowaniu półkrokowym. Przy czym sekwencje sterujące kluczami są takie same jak w naszym modelowym silniku!

A TERAZ PRAKTYCZNIE

Rys. 3 przedstawia schemat ideowy przykładowego tranzystorowego klucza elektronicznego. Podanie wysokiego poziomu logicznego na tranzystor T1 przez opornik 4.7 KOhm spowoduje włączenie klucza i przepływ prądu w obwodzie plus zasilania silnika, uzwojenie silnika, tranzystor T2, masa (minus zasilania silnika). Pozornie niepotrzebna dioda D1 zabezpiecza tranzystory przed przepięciami indukującymi się w uzwojeniach podczas włączania i wyłączania napięcia.

Innym, bardziej eleganckim i prostszym rozwiązaniem klucza elektronicznego jest zastosowanie układu scalonego typu ULN 2003. Nie jest on produkowany w kraju ale widziałem go w ofercie każdej firmy handlującej zachodnimi częściami elektronicznymi. W Warszawie jest np. do dostania w Centralnej Składnicy Harcerskiej.

Układ ten zawiera 7 wzmacniaczy których wyjścia są praktycznie identyczne jak klucz na rys. 3, i szczególnie dobrze nadaje się do współpracy z układami mikroprocesorowymi. Wygodne jest też to, że zawiera diody zabezpieczające przed przepięciami (D1 na rys. 3). Opis wyprowadzeń tego układu znajduje się na rys. 4.

Na rys. 5 znajduje się konkretny przykład rozwiązania sterowania silnikiem krokowym przez interfejs z układem 8255 i układem ULN 2003 jako kluczami. Silnik krokowy zaznaczony jest ciemnym prostokątem. Przewód wspólny jego uzwojeń jest dołączony do plusa napięcia zasilania silnika krokowego. Także do tego napięcia powinien być dołączony plus zasilania układu ULN 2003 (noga 9).

Minus napięcia zasilania silnika krokowego powinien być połączony z nogą 8 układu ULN 2003 oznaczoną "masa" na rys. 4, oraz z masą układu 8255 czyli masą całego komputera.

Uwaga! Silnik krokowy i układ ULN 2003 powinien być bezwzględnie zasilany z odrębnego zasilacza o odpowiednim napięciu i dostatecznej wydajności prądowej!

W opisanym typie silnika z obudowy wyprowadzonych jest 5 przewodów (jeśli są wyprowadzone tylko 4 to jest to silnik bipolarny który steruje się w inny sposób). Jak je rozpoznać? Zajrzeć do katalogu! Jeśli go nie posiadamy to najpierw omomierzem znajdujemy przewód wspólny. Potem dołączamy ten przewód do plusa napięcia zasilania o odpowiedniej wartości i zwierając kolejno do minusa zasilania pozostałe przewody fazowe metodą prób i błędów ustalamy kolejność która powoduje obracanie wirnika w jedną stronę.

Zgodnie z tą kolejnością łączymy przewody fazowe z wyjściami 4 kolejnych wzmacniaczy układu ULN 2003, a wejścia tych wzmacniaczy łączymy też zgodnie z kolejnością z czterema najniższymi bitami portu PA układu 8255.

Jeśli teraz do portu PA wyślemy bajt 00000001 to przez L1 (rys. 5) popłynie prąd. Odpowiada to włączeniu klucza K1 na rys. 2. Wysłanie bajtu 00000010 odpowiada włączeniu klucza K2. Widzimy, że ustawiony ("1") bit PA0 odpowiada włączonemu kluczowi K1, bit PA1 — K2. Odpowiednio bit PA2 — K3 i PA3 — K4. Reszta — to tylko programowanie.

PRZYKŁADOWE PROGRAMY

Listing nr. 1 przedstawia napisany w Basicu ZX Spectrum przykładowy program sterowania silnikiem krokowym połączonym z komputerem przez interfejs jak na rys. 5.

Zmienna RS w linii 5 to adres rejestru sterującego, a PortA to adres portu A. W linii 8 następuje wysłanie słowa sterującego do RS. Ustawia ono wszystkie porty jako Wy. Zmienna WT to wskaźnik tablicy określający który bajt sterujący krokami ma zostać wysłany do portu PA. Linie 10 — 50 to deklaracja tablicy kroków i wypełnienie jej bajtami sterującymi poszczególnych kroków. Można zauważyć że jest to sterowanie pełnokrokowe. W liniach 300 do 500 następuje przypisanie zmiennej ILE żądanej liczby kroków oraz wybór kierunku obrotów. Linie 70 do 90 (170 do 190) realizują zapętlenie wskaźnika tablicy kroków i wysłanie jednego bajtu sterującego do portu PA przy obrotach w prawo (w lewo).

Podstawową wadą tego programu jest jego mała szybkość co powoduje stosunkowo wolne obroty.

Pełnię możliwości sterowania silnikiem krokowym zapewnia program napisany w języku asemblera Z80 z listingu nr 2. Silnik jest sterowany półkrokowo (POTAB to tablica półkroków).

Procedury XLEWO i XPRAWO zmieniają odpowiednio wskaźnik tablicy WSKTBX kroków zapętlając go po dojściu do „skrajnych" wartości. Procedura WYBUF wywołuje procedurę OPOZN, która zapewnia opóźnienie między kolejnymi krokami oraz wysyła bajt sterujący z tablicy kroków do portu PA. Procedura SPRAW rozpoznaje zadany kierunek obrotu i wywołuje odpowiednie wcześniej opisane procedury.

Główną procedurą jest procedura OBROTY (adres startowy 60086) która wywołuje wszystkie pozostałe. Pobiera ona z komórek pamięci o adresach symbolicznych ILEKRO (adres 60000, 60001), KIER (adres 60002) i CZAS (adres 60003, 60004) odpowiednio zadaną liczbę obrotów (0—65535), kierunek obrotów (0-lewo, 1-prawo) i opóźnienie między kolejnymi krokami (200—2000). Należy więc przed wywołaniem tej procedury wpisać do tych komórek żądane wartości.

Przykład korzystania z tej procedury z poziomu Basic'a pokazany jest na listingu nr 3. W linii 10 deklarowane są wartości początkowe zmiennych. Nadając zmiennej LICZBA wartość odpowiadającą liczbie kroków, zmiennej STRONA 0 dla obrotów w lewo lub 1 dla obrotów w prawo, a zmiennej OPOZN wartość z zakresu 200—2000 i wywołując podprogram zaczynający się w linii 9000 możemy sterować silnikiem (linie 100—140).

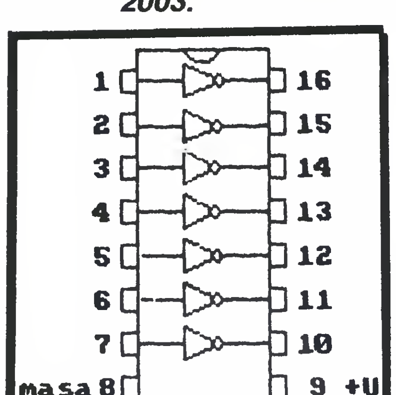
Oczywiście także „czysto" w Basic'u można zrealizować sterowanie półkrokowe. Wystarczy w programie z listingu nr 1 zwiększyć wymiar tablicy do 8, wpisać do niej wartości bajtów sterujących z tablicy POTAB z listingu nr 2 oraz zmienić wartości zmiennej WT w linii 70 z 5 na 9 oraz z 4 na 8 w linii 170. Niestety, silnik nie będzie kręcił się zbyt szybko.

Procedury z listingu nr 2 są częścią programu sterującego dwoma silnikami krokowymi produkcji polskiej typu FA-17-4-1 i elektromagnesem sterującym pisakiem w prostym ploterze wykonanym przez autora wg opisu w niemieckim czasopiśmie Elektor nr 1,2/88.

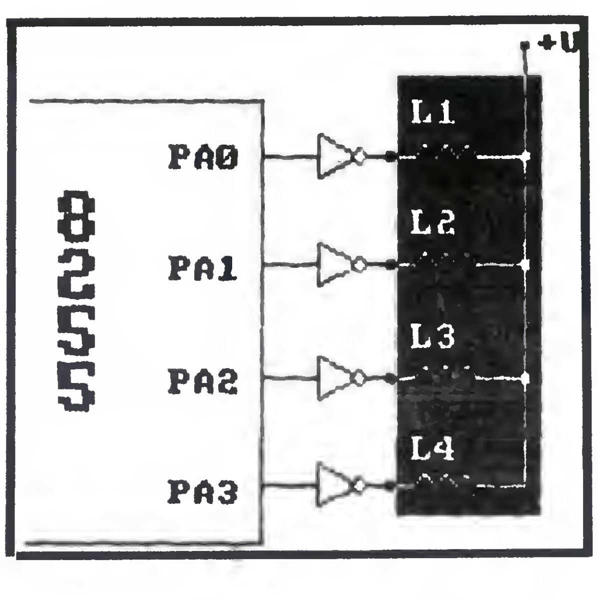
Grzegorz Bujanowski

Literatura:
1. J. Pieńkoś, Układy mikroprocesorowe 8080/8085.
2. J. Pieńkoś, Układy scalone TTL w systemach cyfrowych.

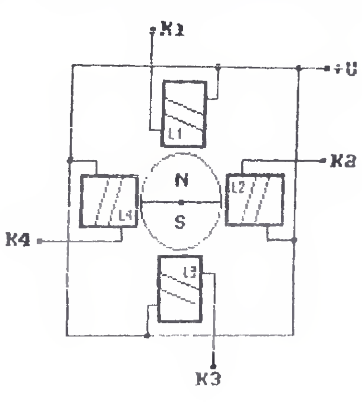
Rys. 4 Opis wyprowadzeń układu ULN 2003.



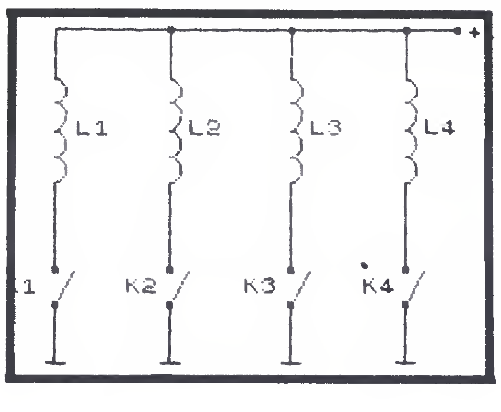
Rys. 5 Schemat podłączenia silnika krokowego z układem ULN 2003.



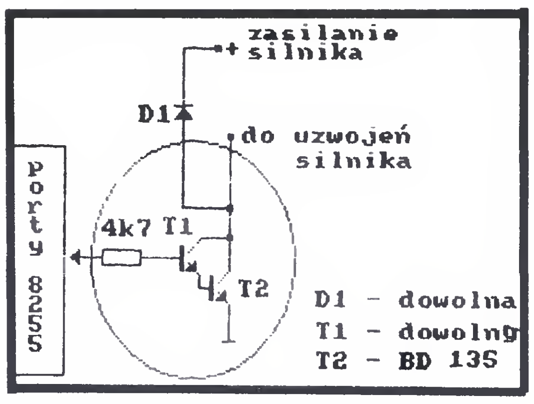
Rys. 1 Model silnika krokowego.



Rys. 2 Schemat ideowy silnika.

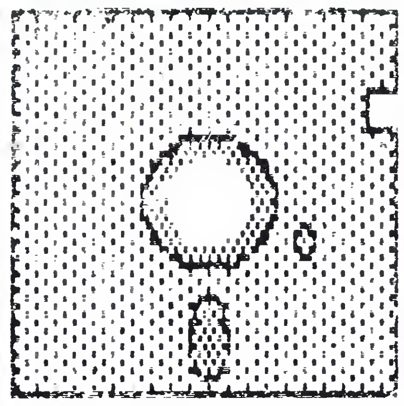


Rys. 3 Przykład klucza włączającego jedno uzwojenie (fazę).



	Giełda	Sklep	Pewex	Zachód
	tys. zł		$	
SINCLAIR				
ZX 81	200	—	—	—
ZX Spectrum 48	800	990	—	—
ZX Spectrum +	900	1100	—	—
ZX Spectrum + 2	1200	—	—	—
Timex 2048	1000	1200	—	—
Stacja FDD 3000	1100	1300	—	—
Stacja FDD 3	800	—	—	—
drukarka GP–50	650	550	—	—
Masterface I	120	—	—	—
AY 3–8910	170	—	—	—
COMMODORE				
C 64	1900	1950	207	—
C 128	2500	2700	—	—
C 128 D	4000	4200	599	—
Amiga 500	5000	5800	795	—
Amiga 2000	12000	—	—	—
Amiga 3000	—	—	—	—
magnetofon	350	400	30	—
stacja 1541	1800	1900	—	—
stacja 1571	2200	—	—	—
stacja Oceanic	1900	—	—	—
drukarka LC 10C	2100	—	299	—
Final II	100	120	—	—
Final III	230	250	—	—
Action Replay	350	—	—	—
ATARI				
Atari 800 XL	1200	1250	—	—
Atari 65 XE	1500	1600	159	—
Atari 130 XE	1900	2200	239	—
Atari 520 STFM	—	4300	499	—
Atari 1040 STFM	—	—	899	—
magnetofon	300	350	51	—
stacja CA 2001	2600	—	249	—
monitor SM124	2200	—	—	—
monitor SM224	—	—	480	—
Turbo 2000	100	100	—	—
Centronics	200	220	44	—
AMSTRAD				
Amstrad 464	2000	2200	—	—
Amstrad 6128	2700	2900	—	—
Amstrad PCW 8256	—	—	—	—
Amstrad PCW 8512	3600	—	—	—
IBM				
IBM PC XT stand.	8000	9500	499	—
IBM PC AT stand.	11000	—	789	—
IBM PS/2	—	16000	—	—
HD 20 MB z kontrol.	3100	—	260	—
napęd 5.25"	600	700	—	—
monitor amber	—	—	219	—
klawiatura	440	—	—	—
INNE				
dyskietki 5.25"	4.5	4.5–8	0.5–3	—
dyskietki 3.5"	15	15–20	1–3	—
dyskietki 3"	29	35–45	—	—
kasety C–60	10	—	1	—
monitor	700	750	—	—
joystick	60–80	60–100	7–11	—

Sklepy „Bajtka" **Bytom, ul. Koniewa 6, tel. (832) 81-49-17**
Kraków, ul. Pstrowskiego 9, tel. (012) 22-59-72



INDYWIDUALNY BANK DANYCH

Marcin Stasiak, lat 15, posiada Atari 65 XE, magnetofon XC 12 z Uniwersal Turbo (Turbo 2000, Blizzard Turbo) oraz stację dysków CA 2001. Nawiąże kontakt w celu wymiany oprogramowania, literatury i doświadczeń.
Adres: oś. Waryńskiego 219/38, 98-300 Wieluń, woj. Sieradzkie.

Wojciech Sterna, lat 15, posiada Atari 130 XE ze stacją dysków 1050, magnetofon z interfejsem. Nawiąże kontakt w celu wymiany doświadczeń i oprogramowania.
Adres: ul. Piotrkowska 12, 61-353 Poznań.

Jarosław Beren, lat 16, posiada Amigę 500 z 1 MB pamięci. Pragnie nawiązać kontakt w celu wymiany oprogramowania i doświadczeń.
Adres: ul. Matejki 14/17, 39-400 Tarnobrzeg.

Ireneusz Wojtaszczyk, lat 15, posiada Atari 65 XE z magnetofonem XC 12 oraz około 150 programów. Nawiąże kontakt w celu wymiany doświadczeń i oprogramowania.
Adres: ul. Sportowa 5, 67-410 Sława, woj. Zielonogórskie.

Robert Rudnicki i Arkadiucz Walczyk, uczniowie kl. VIII, posiadają Commodore 64 z magnetofonem oraz kilkadziesiąt gier. Pragną nawiązać kontakt z początkującymi posiadaczami tego komputera w celu wymiany spostrzeżeń, doświadczenia.
Adres: ul. Poznańska 53 m 66 B, 93-134 Łódź.

Stanislav Figlar, student, lat 20, posiada komputer „SAM Coupe 256 kB + drive 3,5". Nawiąże kontakt z innymi użytkownikami tego komputera w celu wymiany programów i literatury. Korespondencja w języku polskim.
Adres: Tulska 312, 010 08 Žilina, CSFR.

Mariusz Walendowski, lat 16, posiada Timex 2048, magnetofon, około 400 gier i programów użytkowych. Proponuje wymianę oprogramowania.
Adres: 56-400 Oleśnica Śląska, woj. Wrocławskie, Spalice 39 B.

Janusz Bubik, lat 19, posiada ZX Spectrum+, interface Sound. Zainteresowania: elektronika, grafika oraz muzyka komputerowa. Proponuje wymianę programów muzycznych na AY.
Adres: ul. U.výstaviště 3, 750 00 PŘEROV, Czechosłowacja.

Rafał Szempliński, lat 14, posiada Commodore +4, magnetofon 1531 oraz 100 gier. Proponuje wymianę gier, oposów i literatury.
Adres: 10-685 Olsztyn, ul. Baracza 19/14.

Robert Bariasz, lat 16, posiada Atari 800 XL i magnetofon XC12 w systemie Turbo. Oprogramowanie: 200 gier i około 20 programów użytkowych. Nawiąże kontakt z posiadaczami Atari i magnetofonem firmowym z systemem Turbo (dowolny).
Adres: ul. Sienkiewicza 7/19, Łódź.

Piotr Pawelec, lat 18, posiada Atari 130 XE, stację dysków LDN Super 2000 oraz 50 dyskietek z oprogramowaniem. Nawiąże kontakt w celu wymiany doświadczeń i programów.
Adres: ul. Strażacka 27, 44-352 Czyżowice.

Wasilewski Marcin, lat 17, posiada Atari 65 XE oraz stację dysków CA-2001, oraz 250 programów użytkowych i gier. Nawiąże kontakt w celu wymiany oprogramowania i doświadczeń (Turbo Basic XL).
Adres: ul. Dunikowskiego 8a/8, 66-410 Gorzów Wlkp.

KUPIĘ ● SPRZEDAM ZAMIENIĘ

Każdy, kto przyśle do nas dwa wycięte z kolejnych numerów „Bajtka” kupony (odbitek nie będziemy honorować), może zamieścić krótkie ogłoszenie. Maksymalna długość ogłoszenia — piętnaście słów razem z adresem, drobne odchylenia do zaakceptowania. Ogłoszenie może dotyczyć sprzedaży, kupna lub zamiany komputera i akcesoriów — wszelkiego typu urządzeń zewnętrznych, programów i literatury, używanych i nowych, pod warunkiem, że oferta dotyczyć będzie pojedynczych sztuk. Ogłoszenia drukować będziemy kolejno w miarę ich napływania. Zastrzegamy sobie prawo niewydrukowania lub zredagowania ogłoszenia zbyt długiego, anonimowego, a także w razie podejrzeń o próbę sprzedaży hurtowych ilości towaru lub kradzionego oprogramowania, oraz prawo do zmiany zasad akceptowania ogłoszeń. Piszcie na nasz adres, z dopiskiem na kopercie — Kupię-Sprzedam-Zamienię.

1. Sprzedam magnetofon do Spectrum (SVI-767), T. Czyżewski, Łomża, tel. 68-344.
2. Kupię kabel łączący ZX Spectrum+ z magnetofonem MK433. Krzysztof Dęga, 64-970 Piła 4, skr. poczt. 2.
3. Kupię FDD 3000. Konrad Brożyna, 01-494 Warszawa, ul. Wolfkego 16 m 39.
4. Kupię interfejs MIDI z oprogramowaniem do Spectrum+. Andrzej Żywiołek, 73-110 Stargard, ul. Pogodna 30/1.
5. Kupię używane ZX Spectrum+. Adam Turowski, 16-300 Augustów, ul. Śródmieście 1/33.
6. Zamienię Spectrum 48 i interface na trzy joysticki (2 * Sinclair + Kempston) na Timex-a. Robert Heber, 41-910 Bytom, ul. Chorzowska 20/42.
7. Sprzedam Timex-a z firmowym magnetofonem i grami. Ryszard Siech, 90-368 Łódź, ul. Piotrkowska 182/321, tel. 366103.
8. Kupię ZX Spectrum 128. Tomasz Wojna, 59-220 Legnica, ul. Rewolucji Październikowej 22/4.
9. Sprzedam interfejs AY do Spectrum/Timex-a. Krzysztof Karbownirzyn, 21-040 Świdnik, ul. Przod. Pracy 34/8, tel. 140-29.
10. Sprzedam tanio Timex 2048 z FDD 3000 (2 napędy), joystickiem, literaturą, programami. Jerzy Rogowski, Warszawa, tel. 45-41-94.
11. Sprzedam Unipolbrit 2086 (ZX Spectrum) + peryferia. Oferty z ceną. Piotr Gogolin, 42-610 Tarnowskie Góry, ul. Srebrna 8/1/8.
12. Kupię ULA do ZX48. Krzysztof Janicki, 45-707 Opole, ul. Wrocławska 18d/9.
13. Sprzedam magnetofon TIMEX i interface KEMPSTON do Spectrum. Maciej Mocek, 88-100 Inowrocław, ul. Daszyńskiego 7/2.
14. Kupię Timex FDD 3000. Waldemar Miotk, 84-100 Puck, ul. Przebendowskiego 23/38.
15. Sprzedam zestaw: ZX Spectrum+, ZX Printer, monitor TWM-315, 2 joysticki, literaturę, oprogramowanie. Radosław Grzymkowski, 44-117 Gliwice, ul. Kopernika 87/9.
16. Sprzedam AY-3-8910 za 90 tys. zł, lub zamienię na „Sound 128” bez AY. Marek Siłakowski, 87-300 Brodnica, ul. Duży Rynek 15/3.
17. Kupię folię pod klawiaturę i ULA do Spectrum 48. Robert Majewski, 38-200 Jasło, ul. Koralewskiego 2/8.
18. Sprzedam stację dysków Opus Discovery (porty: joystick, Centronics) do Spectrum. W. Kamiński, Łódź, ul. Obr. Stalingradu 11/5, tel. 33-45-70.
19. Nawiążę kontakt z posiadaczami ZX81 celem wymiany oprogramowania. Jacek Szymanowski, 16-400 Suwałki, ul. Emilii Plater 28/4.
20. QL — kupię programy, literaturę, kasetki do microdrive-u. Marek Ciunowicz, Warszawa, 481450.
21. Sprzedam Amstrada CPC 6128, drukarkę DMP2000, dyskietki, literaturę. Jarosław Cecot, Kielce, tel. 434-95.
22. Sprzedam komputer Amstrad CPC 6128, zielony monitor, oprogramowanie, polska instrukcja. Jarosław Radoń, Bielsko Biała, tel. 29374.
23. Kupię gry na Amstrada CPC 464, Bartek Wędołowski, 81-068 Gdynia, ul. Wiejska 7/50.
24. Sprzedam Amstrada 464, zielony monitor, modulator, stacja 5.25", drukarka 42/84. Piotr Stasiński, 02-784 Warszawa, ul. Janowskiego 54/35, tel. 641-63-82.
25. Kupię Amstrada CPC 464 z monitorem mono, programami. Jacek Staszak, 59-820 Leśna, ul. Kochanowskiego 18/2, tel. 266.
26. Kupię symulatory lotu oraz programy muzyczne na Commodore 64. Łukasz Kobiela, 43-430 Skoczów, ul. Szewczyka 23.
27. Kupię oprogramowanie i literaturę o C-64. Michał Kozłowski, 27-400 Ostrowiec Św., ul. Z.W.M. 7 m. 16, tel. 28-649.
28. Sprzedam Commodore 64 z magnetofonem i joystickiem. Mariusz Cheliński, 65-530 Zielona Góra, ul. Zawadzkiego 21/7 tel. 645-99.
29. Kupię pilnie kabel RGB (komputer-monitor C-128) i zasilacz. Dariusz Grochowski, 12-100 Szczytno, ul. Kajki 8.
30. Kupię Commodore C-64 z magnetofonem firmowym. Adam Ogorzałek, 78-100 Kołobrzeg, ul. 1-go Maja 41/3.
31. Kupię trójnik do C-64 na dwa magnetofony. M. Piernicki, 83-110 Tczew, ul. Niepodległości 9F/5.
32. Sprzedam Commodore 16. Odkupię lub zamienię się na wszelką literaturę o C-64 i C-128. Artur Karaźniewicz, 11-111 Kraszewo.
33. Tanio sprzedam Commodore 64C, magnetofon na gwarancji, programy — joystick gratis! Adam Złamaniec, 21-500 Biała Podlaska, ul. Orzechowa 52/19, tel. 43-61-08.
34. Sprzedam Commodore C-16, magnetofon, joystick, zielony monitor. Rafał Pelc, Strzelce Krajeńskie, ul. Wolności 34/9, tel. 763.
35. Kupię Commodore C-64. Marek Czajkowski, 40-048 Katowice, ul. Kościuszki 22/1.
36. Kupię modulator RF do Commodore, P/N 1001027-03 R.O.C. (EUR). Andrzej Tokarek, 49-340 Lewin Brzeski, ul. Moniuszki 29/8.
37. Sprzedam magnetofon C-128/64. Sławomir Żurawski, 71-086 Szczecin, ul. Kmieca 26, tel. 531-666.
38. Kupię obudowę magnetofonu do C-64. Rafał Rychter, 59-300 Lubin, ul. Sokola 8/30.
39. Sprzedam C-64 II, Datasette 1535, Final II, joystick, programy, literaturę, cena 240 $. Paweł Bielecki, Warszawa, tel. 58-80-65.
40. Kupię Interface do magnetofonu C64. Mikołaj Dreżewski, 80-809 Gdańsk, ul. Cebertowicza 18/28.
41. Sprzedam drukarkę Epson LX-400, interfejs Centronics C-64, C-128. Józef Ostaszewski, Legnica, tel. 202-17.
42. Kupię komputer Amiga 500 i oprogramowanie. Lucjan Danek, Rzeszów, ul. Krakowska 159.
43. Odsprzedam oryginalne programy na Amigę: Kindwords, FusionPaint, F-18 Interceptor. Aleksander Buczkowski, 42-609 Tarnowskie Góry, ul. L. Solskiego 73, tel. 85-87-11.
44. Kupię oprogramowanie na Amigę 500. Adam Stanek, 25-370 Kielce, ul. Żeromskiego 44/56.
45. Kupię IBM PC. Artur Motyl, 21-041 Minkowice, Krępiec I 68A.
46. Kupię używanego PC/XT. Mateusz Kwinta, 30-003 Kraków, ul. Lubelska 7/2, tel. 33-09-20.
47. Sprzedam Amstrad-Schneider PC1512SD, 512 KB, FDD 5.25" 720 KB, kolorowy monitor, mysz za 9.5 mln zł. Marek Łach, 32-522 Jaworzno, ul. Partyzantów 20/20.
48. Kupię HDD+kontroler XT, monitor do Herculesa, Ryszard Trojnara, 21-110 Ostrów Lubelski, ul. Armii Lud. 3c/8.

Drogi Bajtku!

Na listy Czytelników odpowiadają autorzy „Bajtka”

Jestem stałym czytelnikiem „Klanu Amstrad”. Próbowałem wprowadzić na swój użytek proponowane przez Was rozszerzenia RSX, ale bez zadowalającego efektu. Wykonuję asemblację programem RMAC z dyskietki systemowej. W artykule o rozszerzeniach RSX proponuje się użycie programu M80, którego ja do tej pory nie spotkałem na żadnych dyskach dostarczanych z komputerami Amstrad PCW.

Jeżeli powodem moich niepowodzeń są, jak podejrzewam, różnice między programem RMAC a M80, to prosiłbym o wyjaśnienie tych różnic i podanie źródła, skąd można uzyskać asembler M80, lub uświadomienie mi, gdzie popełniam błąd. Procedury nie używające odwołań do asemblera nie nastręczają kłopotów.

Grzegorz Pawełka

Wydaje mi się, że Pańskie problemy spowodowane są różnicami między programami RMAC i M80. Główna różnica polega na tym, że M80 jest bardziej rozbudowany, daje możliwość pisania w asemblerze Z80 i Intela 8080 (RMAC — tylko Intel 8080) oraz posiada wiele dodatkowych opcji i pseudorozkazów. Większość tych opcji jest ignorowana lub źle interpretowana przez RMAC-a.

Niestety, M80 jest produktem firmy MicroSoft (a nie Digital Research), co oznacza że nie był, nie jest i nie będzie dostarczany na firmowych dyskach z systemem CP/M. Można go zakupić na Zachodzie (drogo, ale legalnie) lub na giełdach (taniej, ale nielegalnie).

(M.Sz.)

Od niedawna jestem posiadaczem C64 z magnetofonem i modułem PLUS. Mam w związku z tym kilka pytań:

1. Zakupiłem cartridge o nazwie PLUS. Niektóre programy nie działały, gdy PLUS był załączony. Gdy ponownie załadowałem grę bez modułu, uruchomiła się bez problemu.
2. Po włączeniu komputera klawisz F1 działał jak spacja, a SPACE nie działał w ogóle.
3. Czy magnetofon po przewinięciu taśmy może się sam wyłączyć?
4. Zdarza się, że po informacji PRESS PLAY ON TAPE i uruchomieniu magnetofonu silnik nie rusza.

Jakub Trzaska
Sosnowiec

1. Zdarzają się programy, które nie działają w obecności modułów rozszerzających. Związane jest to z umieszczeniem procedury startującej program w obszarze pamięci zajmowanym przez cartridge. Moduły takie jak np. FINAL II można wyłączyć komendą kill, a cartridge X (POWER! MAGIC) po prostu wyłączyć wbudowanym przełącznikiem. Najprostszym jednak sposobem jest po prostu wyjąć moduł i załadować program 'na piechotę'.
2. Powodem zaistniałej sytuacji mógł być źle włożony w port moduł lub źle zresetowany C64.
3. Magnetofon sam potrafi wyłączyć jedynie klawisz PLAY.
4. Takie zachowanie się magnetofonu może być spowodowane sytuacjami:
— kaseta skończyła się i nie została przełożona na drugą stronę,
— wtyk magnetofonu jest źle włożony,
— komputer jest źle wykasowany.

(TNL)

W imieniu posiadaczy stacji dysków ze Świnoujścia proszę o prawidłowe wydrukowanie programu „Multi DOS” lub erraty do niego [Program był publikowany w „Moim Atari” nr 2].

Marcin Sowa
Świnoujście

Do listingu programu „Multi DOS” zakradło się kilka błędów. Bardzo przepraszamy.

Liczby: 9, 175, 1181 znajdujące się pomiędzy liniami 146 i 255 stanowią zakończenie linii 254.

Liczby: 114, 101, 992 znajdujące się pomiędzy liniami 281 i 390 stanowią zakończenie linii 389.

Liczby: 64, 172, 1344 znajdujące się pomiędzy liniami 416 i 174 stanowią zakończenie linii 173.

Liczby: 3, 209, 1142 znajdujące się pomiędzy liniami 254 i 147 stanowią zakończenie linii 146.

Liczby: 115, 107, 931 znajdujące się pomiędzy liniami 173 i 417 stanowią zakończenie linii 416.

Liczba: 590 znajdująca się pomiędzy liniami 428 i 336 stanowi zakończenie linii 335.

Liczby: 133, 209, 1464 znajdujące się pomiędzy liniami 362 i 282 stanowią zakończenie linii 281.

Liczba: 74 znajdująca się pomiędzy liniami 308 i 363 stanowi zakończenie linii 362.

Liczby: 9, 32, 1137 znajdujące się pomiędzy liniami 389 i 309 stanowią zakończenie linii 308.

Oczywiście kolejność wprowadzania linii Basicowych jest nieistotna!

(wist)

P.U „FORMAT"

01–031 Warszawa, ul. Marchlewskiego 59/73
tel. 38–07–76
oferuje:

Zewnętrzne Stacje Dysków

wszelkich typów (5,25", 3,5", 3")
do komputerów domowych, przenośnych, profesjonalnych.

Amiga Atari ST, Amstrad. Schneider, Toshiba Bondwell, Spectrum, PS/2, XT, AT i innych.
oraz
Rozszerzenia pamięci do Amigi

B92

„MARK SOFT"
ATARI XL, XE

programy użytkowe, edukacyjne, gry,
instrukcje opisy gier, podręczniki.
Wysyłka na cały kraj.
Przeróbki sprzętowe.

TURBO 2000

informacja: koperta + znaczek
„MARK SOFT"
ul. Kasprowicza 88 m 103
01-949 Warszawa

B100

ATARI 800 XL, 65 XE, 130 XE

Sprzedaż wysyłkowa
gier i programów użytkowych
na kasetach i dyskietkach.
Również w systemie TURBO 2000
Wszystkie nowości!!!
Instrukcje i literatura.
Dla zainteresowanych rachunki.

ANWIKOL

03-721 Warszawa ul. Jagiellońska 3/28.

B81

KOMPUTER
NATYCHMIAST
KUPISZ-
SPRZEDASZ

MAXSOFT

659-44-17 Warszawa

B74

ZX SPECTRUM ATARI
system turbo

TIMEX FDD 3000 AUDIO — VIDEO

— programy użytkowe, edukacyjne, gry
— instrukcje, podręczniki
— kasety muzyczne (ponad 300 tytułów)
— kasety video (filmy, bajki)
— kasety audio i video-czyste
— wysyłka na cały kraj
— rachunki
— informacje po nadesłaniu koperty zwrotnej + znaczek.

„P.K.T.S." Studio Komputerowe
00-103 Warszawa
ul. Królewska 43 m 25
tel. 20 51 25

B79

MIKRO SERWIS

poleca naprawy mikrokomputerów i peryferii

Specjalna oferta:

- rozszerzenia RAM do Amigi 500:
  — 512 kB
  — 1,8 MB
- cartridge do C-64

Dla zamiejscowych naprawy na poczekaniu.

Gdańsk, ul. Marusarzówny 6
tel. (058) 48-50-63

B94

Atari Turbo 2000 F:

Nowy system transmisji danych z magnetofonem przyspieszony do 6700 bodów.
Komplet:
— cartridge
— oprogramowanie
— przeróbka magnetofonu
— instrukcja obsługi
— 12 miesięcy gwarancji.
Instalacje wykonujemy na poczekaniu.

Interfejs do zwykłego magnetofonu.
Duży wybór oprogramowania w standardzie TURBO-2000.
Informacja:
Tel. 33-40-91
Korespondencja i wykonywanie usług:
MUEL ul. Cząstkowska 30
01-678 Warszawa.

B82

ATARI XE, XL SPECTRUM, TIMEX
NISKIE CENY PROGRAMÓW NA TAŚMIE I DYSKU
INTERFEJSY TURBO I „AY" DO SPECTRUM
CARTRIDGE DO ATARI — GRY I UŻYTKI
INFORMACJA — ZAADRESOWANA KOPERTA + ZNACZEK
05-220 ZIELONKA SKR. POCZ. 9/2

B80

AMIGA, C-64
ATARI XL, ST
Programy za zaliczeniem pocztowym
katalogi — gratis
co 5 program — gratis
szybko, tanio oferuje:
Andrzej Zygmunt tel. 588–733
ul. Owocowa 4/9,
40-158 Katowice

B89

Studio Komputerowe „PomareX" poleca programy dla komputerów:

ATARI XE/XL, COMMODORE, AMIGA.
Informacja na miejscu lub za załączeniem koperty zwrotnej.
Lublin, ul. Sławińskiego 12

B101

Zakład Usług Elektronicznych „HOMECOMP"

(do niedawna AZUSPHW) poleca usługi w zakresie serwisu komputerów: Spectrum, C-64, C+4, Timex, Atari oraz zasilaczy komputerowych.
Warszawa ul. Puławska 102,
tel. 44-87-89
czynny w godz. 11 — 19, rachunki, gwarancja.

B95

SOUND

— trójkanałowy, stereofoniczny, przelotowy interfejs muzyczny (AY – 3 – 8910) do ZX Spectrum i Timexa.
Możliwość wysyłki pocztą.
„DYMAREX"
ul. Meissnera 14 m 1,
03–982 Warszawa
tel. 15-93-38 godz. 18—20

B111

Loader do gier dla Atari współpracujący ze stacją XF 551. Umożliwiający szybki (Turbo) odczyt z obu stron dysku jednocześnie. Cena 50 000,- Zamówienia (realizowane za zaliczeniem pocztowym) kierować na adres:

Jacek Szmyd
Wiosenna 15
35-303 Rzeszów

B107

Programy C-16, C-116, C PLUS 4 C-64 wysyłam pocztą. Katalog gratis po otrzymaniu koperty zwrotnej.

Nagrywanie programów Komputerowych
ul. Lenina 104/3
58-304 Wałbrzych

B77

JOLA

02-117 Warszawa
ul. Racławicka 144 m 112
tel.56-00-07 godz. 8-15

ATARI XE/XL

- Gry i programy użytkowe
- Programy kasetowe i dyskowe
- Gry w TURBO 2000
- Opisy, instrukcje i literatura

SZEROKA OFERTA PROGRAMÓW

- TOP DRIVE 1050
- TURBO 2000
- Rozszerzenia pamieci
- Przeróbki magnetofonów i stacji dysków
- Cartridge BASIC XE, XL,
- ACTION
- Bezpłatny katalog B-85

Najniższe ceny, zniżki
Zadzwon lub napisz

DZIAŁ REKLAMY
magazynów komputerowych:

BAJTEK
MOJE ATARI
TOP SECRET

przyjmuje reklamy
codziennie w godz.
9.00 — 16.00
tel. 21-12-05

Wszystko do komputerów ATARI

ATARI SUPER TURBO
PIERWSZY I NAJLEPSZY SYSTEM TURBO DO ATARI

- Nowe programy narzędziowe do systemu AST AST Monitor, AST Emulator, Head Test, itd
- Rewelacyjny nowy cartridge z "narzędziami"
- Autoryzowany montaz, serwis, pełna obsługa
- Wszystkie programy w AST, takze nowości !

ATARI XL/XE, ST

Polecamy pełny wybór programów, literatury i instrukcji do wszystkich 8-io i 16-to bitowych komputerów ATARI. Zapraszamy do współpracy!

W KAŻDEJ SPRAWIE I CHWILI

ATARI STUDIO
W-wa ul. Jóźwiaka 4
10⁰⁰-18⁰⁰ ☎ 125-123

Prosimy o kontakt wszystkich piszących programy na ATARI

B103

Studio Komputerowe

EXPRESS

poleca:
— oprogramowanie
— literaturę
— kartridże
dla

ATARI, COMMODORE

napisz!
CO PIĄTY program GRATIS!
programy niezabezpieczone
Błyskawiczna realizacja zamówień
Atrakcyjne zniżki, katalogi
zapamiętaj ten adres:
ul. Romantyczna 15/21
20-533 Lublin
Tylko u nas, wszystko dla Was!
B102

ATARAX

Sprzedaż wysyłkowa
Katalogi gratis: po wysłaniu zaadresowanej koperty zwrotnej + znaczek.
oprogramowanie, literatura, instrukcje

ATARI XL/XE
ATARI ST
COMMODORE C-64
COMMODORE 16, 116, +4
AMIGA, IBM PC XT/AT

ATARAX
05-100 Nowy Dwór Mazowiecki
ul. Chemików 7/15
tel. 75 22 47
B73

MICROMAN

oferuje na miejscu lub wysyłkowo:

1. Sprzęt komputerowy i akcesoria.
2. Programy i literaturę do komputerów: Atari XL/XE/XT, Commodore 16/116/+4/64/ /125/ Amiga, Spectrum, Timex.
3. Oprogramowanie na cartrig'e
4. Przystawki „UNIVERSAL TURBO" do magnetofonów firmowych Atari umożliwiające zapis i odczyt programów w systemie Blizzard oraz Turbo 2000
5. Naprawy zasilaczy, magnetofonów, klawiatur w komputerach Atari, Commodore, Spectrum.

Informacje: (na miejscu lub koperta zwrotna), **zamówienia:**
— 40-181 Katowice ul. Osikowa 66 tel. 585-106
— 44-220 Rybnik ul. Wiejska 10 tel. 233-56

Firmowe punkty sprzedaży:
— Katowice ul. Plebiscytowa 31
— Rybnik D.H. „Hermes" I piętro
— Bielsko-Biała Plac Wolności 9

B88

REKLAMUJ SIĘ W BAJTKU!

Serwis Komputerów

TEST

Katowice, ul. Armii Czerwonej 22/53 tel. 5983 (superjednostka) IX piętro
poleca naprawy:
- ATARI 600, 800, 65, 130 XL, XE
- COMMODORE 16, 116, +4, 64, 128, 128D, AMIGA
- DISK DRIVE 1541, 1570, 1571, 1050 IBM XT, AT

rozszerzenie pamięci:
- ATARI 600 XL, COMMODORE 16, 116, do 64 KB
- ATARI 800 XL, 65 XE, do 130 KB
- AMIGA 500 do 1 MB
- ATARI 520 ST do 1 MB

godz. 9–11, 15–18
B104

ATARIX

ATARI 800 XL
65XE 130XE
literatura
oprogramowanie
katalogi gratis
nizawodność
Warszawa 04-357
Grochowska 186 m 69
B51

PUBLIC DOMAIN SOFTWARE

DLA PC/XT/AT/PS2 PROSTO Z USA

Obejmuje ponad 2000 dysków z tysiącami programów takimi jak: bazy danych, edytory tekstów, programy kalkulacyjne, finansowe, antywirusowe, systemowe, komunikacyjne, języki programowania, grafikę, gry i wiele innych.

CENA DYSKU 5 1/4" z programami — 37000 zł
CENA KATALOGU 2 dyski 5 1/4" — 16000 zł
Powyżej 5 dysków — 10% rabatu
Powyżej 10 dysków — 20% rabatu

NASZ ADRES:
PUBLIC DOMAIN SERVICE
Skrytka poczt.1172
40-001 Katowice 1
tel. 512-109
B72

joy

Wysyłka natychmiastowa za zaliczeniem pocztowym

Joysticki do Atari, Commodore, Spectrum, Amstrad. Precyzyjny mechanizm, specjalne styki.

Kable z wtyczką, przedłużacze do joysticków.
Interface do Spectrum.
6 miesięcy gwarancja.

Elektromechanika
ul. Cegielniana 17
32-410 Dobczyce
B97

•ATASERW•

43-100 Tychy
ul. Lencewicza 46/3
tel. 27 69 66

oferuje świetne rozwiązania sprzętowe

do ATARI XL/XE

1. Turbo Dos — wspaniały Dos na kartridżu
2. Top Drive — do stacji 1050, LDW 2000, California samodzielny montaż — (rec. Informik III/88).
3. Interfejs Centroniks
4. Basic XE — kartridż
5. Tur Dos + Bug65 + Mac65 — kartridż

12 miesięcy gwarancji. Informacje i zamówienia tel. środa i czwartek 16 – 18, listownie po otrzymaniu koperty zwrotnej.
B105

TOS A4

DLA TIMEX FDD 3000
-320 KB PAMIECI NA DYSKU
-WBUDOWANE NAJLEPSZE UZYTKI
-RAMDYSK W STACJI

OFERUJA

J.SZARSKI
KILINSKIEGO 27/17
96-300 ZYRARDOW

R.CYMER
SZAFERA 1/42
92-306 LODZ

B 113


SV 119
Junior
2 Fire
6 Blaszanych styków
Prosty mechanizm
SV 120
Junior–Stick
2 Fire
6 Blaszanych styków
Uchwyt pistoletowy
SV 122
Quickjoy II
2 Fire
6 Blaszanych styków
AutoFire
Drążek lotniczy
SV 124
Turbo
6 Mikrostyków
AutoFire
Drążek lotniczy
SV 123
Supercharger
2 Fire
6 Mikrostyków
Ergonomiczna budowa
Precyzyjny mechanizm
SV 126
Jet Fighter
2 Fire
6 Mikrostyków
AutoFire
ACS-Regulator
szybkości AUTO
Obsługa pod kciuk"
Drążek lotniczy
SV 125
Superboard
6 Fire
10 Mikrostyków
AutoFire
Cyfrowy wyświetlacz
czasu
Sygnał dźwiękowy
Przełącznik dla
leworęcznych
Drążek lotniczy
SV 130
IR Infrared
1 Fire
5 Mikrostyków
Podczerwień
Daleki zasięg
Odbiornik
SV 128
Megaboard
4 Fire
10 Mikrostyków
AutoFire
6 cyfrowy stoper
ATM — Anti Tilt Mechanism
Fire Pad
SV 140
Enterprice
2 Fire
6 Mikrostyków
AutoFire
ACS — Regulator
szybkości AUTO
Drążek lotniczy
„kierownica"
Kabel 4 m
SV 201
Quickjoy M 5
Do IBM XT/AT
(kompatybilnych)
Współpracuje z Game-Card
lub I/O Card
2 Fire
2 AutoFire
6 Mikrostyków
Wybór AUTO
PSC — Regulator XY
Sygnalizacja Świetlna
Fire
ASC — Regulator szybkości
AUTO
SV 202
M 6 analog
Analogowy
DO IBM XT/AT
(kompatybilnych)
Współpracuje z Game-Card
lub I/O Card
2 Fire
SV 210
Game Card
Do IBM XT/AT
(kompatybilnych)
Doskonale pracuje
z M 5 i M 6
SV 127
Top Star
2 Fire
6 Mikrostyków
AutoFire
Przeźroczysta obudowa
SAS — Shock Absorbing
System
Platynowane części
SV 500
Van 3
Pudełka na dyskietki
80 sztuk 3½"
Zamknięcie na klucz
SV 510
Van 5
Pudełko na dyskietki
80 sztuk 5¼"
Zamknięcie na klucz
Quickjoy
TAL — najtaniej w Polsce!

BUSINESS W KOMPUTERZE

(tragedia binarna)

Wśród redakcyjnej poczty największą część stanowią prośby o poradę przy zakupie lub zamianie komputera. „Drogi BAJTKU"! Zamieniam AMIGĘ na MEGA ST [...]; „Mam Spectrum i chcę używać programów typu AutoCad [...]" — to typowe przykłady takich listów. Bardzo rzadko jednak trafia się list, w którym istnieje właściwy porządek — najpierw określam swoje cele, do jakich zamierzam stosować komputer, a dopiero później wybieram sprzęt.



Efekty odwrotnego postępowania wyśmienicie przedstawił R.L. Baber w książce „O programowaniu inaczej", wydanej przez Wydawnictwa Naukowo-Techniczne (ISBN 83-204-1072-X). Spróbujmy przeanalizować prawidłowy i błędny system postępowania.

Pan Bańbuła postanowił założyć własną firmę handlowo-usługową „WSZYSTKONARAZ Sp. Z O.O.". Firma ta ma zajmować się handlem detalicznym i zarządzaniem małym składem celnym, jednakże w bardzo wąskim zakresie i za pomocą rewelacyjnej metody gwarantującej maksymalne zyski i minimalne podatki. Pan Bańbuła doskonale wie, jak komputer ułatwia (ale i utrudnia!) życie, lecz nie zna się na tym dostatecznie, z czego zdaje sobie sprawę (plus).

W pierwszym odruchu pan Bańbuła wpadł na pomysł udania się do swojego serdecznego przyjaciela Ekraniaka, cierpiącego jako fanatyk komputerów na głód klawiaturowy. Pan Bańbuła wie, że Ekraniak jest wyjątkowy, jeśli chodzi o łamanie gier, nie wie jednak, czy jest on w stanie napisać znacznie poważniejszy i trudniejszy program (drugi plus).

Pan Bańbuła wpada więc na lepszy pomysł: opiszę w normalnym języku swoją koncepcję, wszystkie powiązania pomiędzy programami zgodnie z moją metodą i udam się na konsultację do specjalistycznej firmy. Ekraniak co prawda odradzał, bo to niby duże koszty, ale co mi szkodzi...

Zlecenie pan Bańbuła sformułował następująco:
„Nie znam się na komputerach. Programy mają wykonywać następujące czynności (tu lista). Zależności pomiędzy nimi muszą być następujące (następna lista). Proszę o:

a) precyzyjne określenie moich potrzeb sprzętowych,
b) przełożenie moich założeń i potrzeb na język zrozumiały dla programistów,
c) dobór najbardziej efektywnego języka i/lub oprogramowania dla moich potrzeb,
d) określenie dodatkowych opcji oprogramowania nie uwzględnionych przeze mnie, a możliwych do zrealizowania,
e) dokładny opis metod i algorytmów, jakich ma użyć programista podczas projektowania mojego systemu,
f) zaprojektowanie systemu przystosowanego od początku do jego znacznego rozszerzenia w przyszłości w każdym z interesujących mnie zagadnień.

Przez *n* tygodni konsultanci biedzili się nad wymyśleniem najbardziej optymalnego rozwiązania dla pana Bańbuły; analizowano wiele różnych teorii, algorytmów, metod programowania, optymalnych struktur baz danych itp. Założenia zawarte na kartce wyrwanej z zeszytu zajęły teraz opasłe tomisko.

Po *n* miesiącach pan Bańbuła, uboższy o kilkanaście milionów, udał się do firmy produkującej oprogramowanie na zamówienia. Oczywiście wynikły na początku pewne tarcia, gdyż programiści tejże firmy mieli zupełnie inne zdanie na temat potrzeb pana Bańbuły; ponieważ jednak zleceniodawca twardo obstawał przy założeniach opracowanych przez firmę konsultacyjną, programiści skapitulowali i zamówienie przyjęli, podnosząc jednak nieco jego cenę.

Po kilku miesiącach komputer wraz z oprogramowaniem znalazł się nareszcie u właściciela. Zastrzeżenia umieszczone przezornie w kontrakcie i konsultacja wykonanego oprogramowania zmusiły twórców systemu do usunięcia szeregu drobnych błędów oraz efektów lenistwa i niedbalstwa, poprawienia wykonanej dokumentacji systemu i zmiany algorytmów czterech procedur jako niezgodnych z założeniami. Ostatecznie program został przyjęty dopiero po zapewnieniu konsultantów, że system jest CAŁKOWICIE zgodny z ich założeniami. Wykonanie tych operacji nie kosztowało Bańbuły nawet złamanego szeląga, gdyż wszystko to było obwarowane odpowiednim kontraktem, przygotowanym również przez konsultanta.

Wdrożenie systemu i wprowadzanie danych przebiegło bez większych problemów i zajęło dalsze dwa miesiące. System zaczął działać. Po roku pan Bańbuła zlecił jednocześnie: rozbudowę systemu i analizę ekonomiczną zysków, jakie przyniosło mu rozważne postępowanie w nieznanej mu dziedzinie. W rezultacie dowiedział się, że koszty włożone w konsultacje, sprzęt i oprogramowanie zwróciły się już po 7 miesiącach pracy systemu. Ponadto jego rozbudowa przebiegała bez zakłóceń, choć tymczasem pan Bańbuła sześciokrotnie powiększył swój skład celny i rozwinął handel na skalę ogólnokrajową.

Po kilkunastu miesiącach Ekraniaka o mało jasny szlag nie trafił, gdy zobaczył Bańbułę wysiadającego z nowiutkiego mercedesa. Po głębszym namyśle Ekraniak wymyślił jeszcze lepszą metodę od stosowanej przez byłego serdecznego przyjaciela Bańbułę; metoda ta dawała zyski większe o 45%.

Aby nie tracić czasu i pieniędzy, Ekraniak nabył bez zastanowienia najnowszy model IBM PC 386, skopiował od zaprzyjaźnionego księgowego Pałuchy standardowe oprogramowanie i w ciągu dwóch miesięcy rozpoczął działalność. Niestety sprawy od początku szły nie najlepiej. Po pierwsze, zawartość komputera różniła się w sposób znaczny od zawartości samego magazynu (na skutek nie przemyślanej do końca struktury głównej bazy danych). Faktury system drukował poprawnie, zdecydowanie jednak nie chciał drukować nazwisk odbiorców zaczynających się na literę Z, ze względu na wadliwe indeksowanie zbioru danych. Ilość zakupionych za granicą telewizorów zgadzała się wszędzie co do joty, czego jednak nie można już było powiedzieć o magnetowidach, regularnie sumowanych z magnetofonami (zastosowany klucz indeksowy rozróżniał jedynie trzy pierwsze litery nazwy sprzętu). Co kończył się jeden problem poprawiany na bieżąco przez Ekraniaka, to natychmiast wyłaził nowy. O rozszerzeniu działalności nie mogło być mowy — już obecna sprawiała wystarczająco dużo kłopotów. Brak zasilania i brak rezerwowych kopii danych pozwolił Ekraniakowi spędzić urlop na wstukiwaniu ich od nowa.

Po roku pan Ekraniak, wykończony ciągłym poprawianiem i przystosowywaniem programów, poprosił o konsultację pociotka szwagra — programistę Gwizdałę. Po przeprowadzonych badaniach ten ostatni wykrył szereg innych poważnych błędów, wynikających przede wszystkim z zastosowania złych, nieprzemyślanych i błędnie dobranych do metody Ekraniaka algorytmów. Koszty doprowadzenia istniejącego oprogramowania do stanu podstawowej używalności Gwizdała wycenił na sumę prawie dwa razy większą od tej, którą zapłacił były przyjaciel Ekraniaka — Bańbuła. Pan Ekraniak skończył na zawał w karetce reanimacyjnej (o gorzka ironio!) marki mercedes, gdy ujrzał Bańbułę zajeżdżającego tym razem BMW serii 700, kupionym dla nieletniej córki.

Być może zapytasz, Czytelniku, co ta historia ma wspólnego z wyborem komputera. Owszem, ma, i to dużo. Po pierwsze, przemyślany do końca zakup zawsze przyniesie więcej pożytku aniżeli szkody. Chcesz kupić AMIGĘ 2000 ze względu na świetny dźwięk, wspaniałą grafikę i ewentualną możliwość dołączenia karty emulującej IBM? A może lepiej kupić w to miejsce po prostu IBM z kartą graficzną VGA i kartą MT32 firmy Roland zamieniającą komputer w profesjonalny syntetyzer? Drugi przykład: piszę program edukacyjny z matematyki w zakresie układu równań z dwiema niewiadomymi. Do tego nie potrzebuję IBM, wystarczy mi Spectrum. To cel ma uświęcać środki, a nie środki cel!

Odwrócenie tej kolejności prowadzi w wielu wypadkach do nieporozumień i znacznych strat finansowych. Jeśli chcemy odkręcić śrubę, to wiadomo, że potrzebne będą obcęgi; w przeciwnym wypadku trzeba zadać sobie pytanie: „Co mogę zrobić obcęgami?" Przyjemnych rozważań.

Klaudiusz Dybowski